I/O News

MenuWare: A Software Review

Using C-10 "Computer" Ports

Set System Clock with dBASE II

THE OFFICIAL PUBLICATION OF THE INTERNATIONAL ASSOCIATION OF CROMEMCO USERS

Volume Three, Number Five

Single Copy Price $7.50

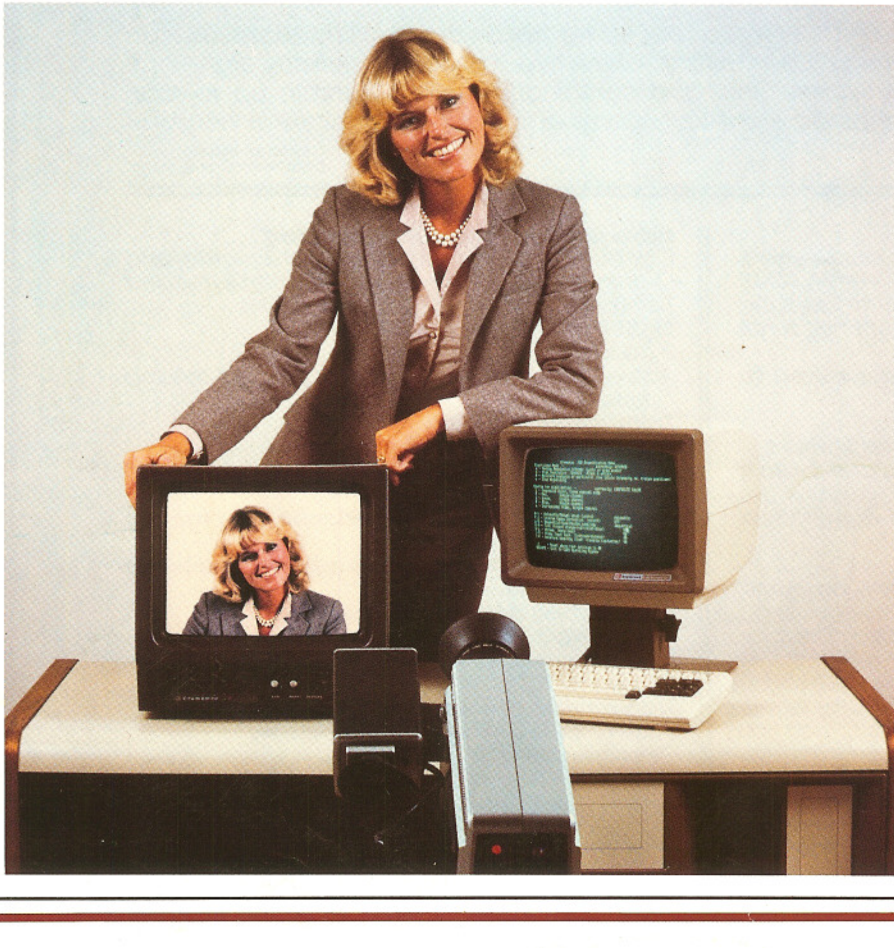
New TV Camera Interface for Cromemco Systems

As the latest addition to its family of graphics hardware and software products, Cromemco has introduced the SDD video digitizer board.

Interfaced directly with standard television cameras, the SDD allows the user to digitize and store full color video images of up to 484 (vertical) by 754 (horizontal) resolution. The output of the SDD can be stored on disk or in memory or could even be transmitted through a serial interface to another machine. Images handled in this way can then be placed in graphics memory for display, modification, or enhancement.

Very often the job of creating a complex image can take hours of skilled operator time, even with an extensive library of programming tools and access to sophisticated application programs, such as the SlideMaster paint system. Furthermore, some images (e.g., radar, medical scans) simply cannot be

Continued on page 8



CROMIX — A User's Guide

by Leigh R. Thomas

I have been a Cromemco user in Australia for several years, and a CROMIX user since that operating system was first released here. I have found, as I am sure many other CROMIX users have, that CROMIX is an excellent system once one learns to use its full potential, but it is rather difficult to learn that potential from the manual.

Certainly the manual is a good, quick-reference guide after one is familiar with the system, but it is not basically tutorial in nature. It is for this very reason that most newcomers to CROMIX have, in my opinion, experienced only a fraction of the potential of the operating system.

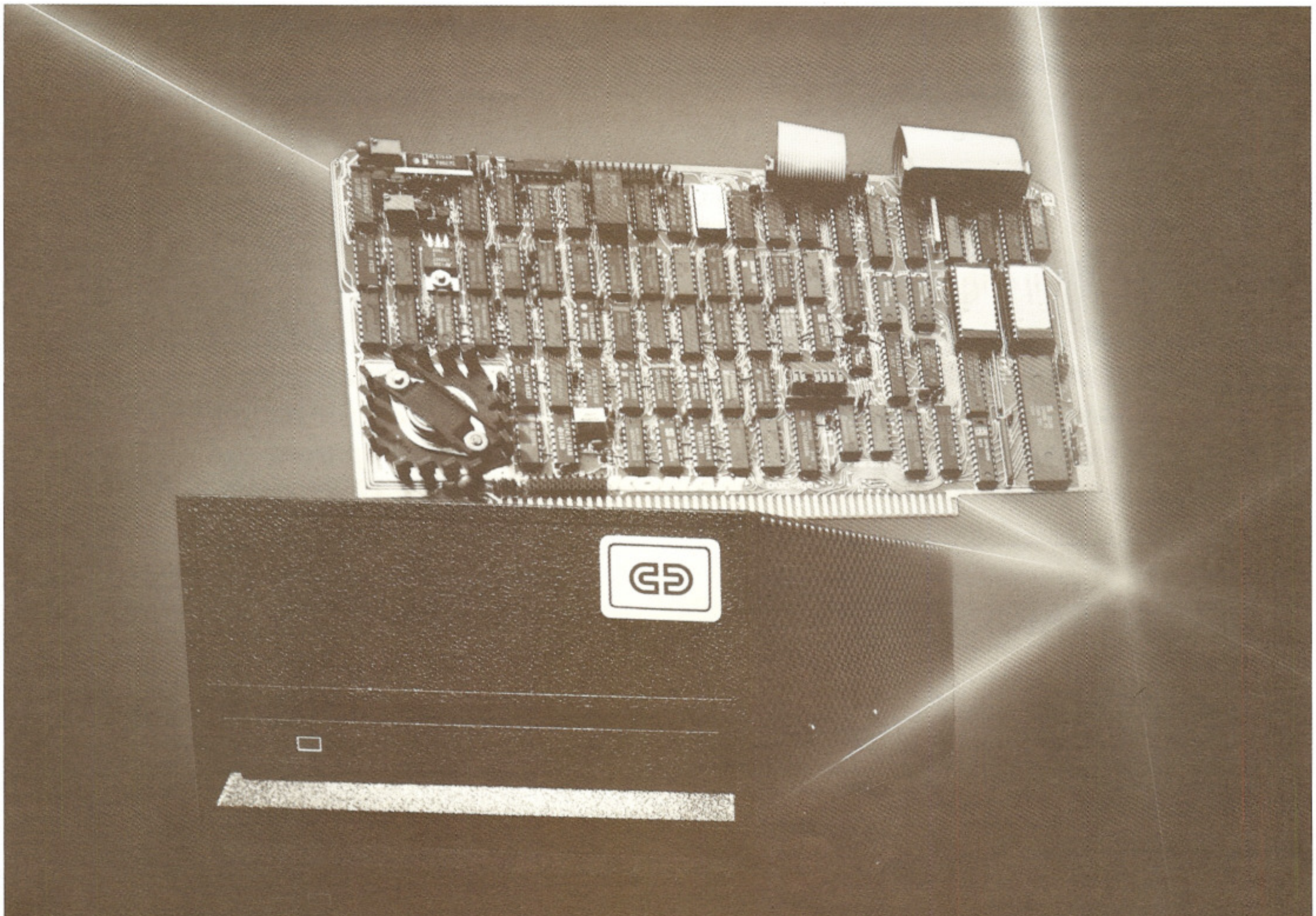
Continued on page 27

Update on Cromemco

CROMEMCO 1984: An Interview with Dr. Harry Garland

In the first issue of I/O News, a little more than three years ago, we ran a feature entitled, **The Cromemco Story.** At that time, Cromemco was the undisputed leader in expandable, bus-oriented microcomputer systems for professional users. Where does Cromemco stand today, and what prompted this interview? The second part of the question is easy to answer. The article is in response to IACU Members who have seen the entire microcomputer industry go through a period of upheaval and subsequent

Continued on page 16



The Hard Disk That's Hard To Believe

35 Megabytes/35 Milliseconds

Introducing the Performance Solution for your Cromemco CROMIX System 1,* System 2 or System 3.* 35 Megabytes of hard disk storage with 35 millisecond average access time.

Complete Package Includes:

- Control Data Corporation Hard Disk
- Disk Controller
- System Generation program
- Extensive Disk Diagnostics
- Efficient disk I/O drivers provide additional shell buffers
- Custom CROMIX I/O and Utilities
- One Year Warranty (EXPRESS Maintenance Service Available)

This new 35 Megabyte Hard Disk Subsystem is available only from **Gunn Enterprises, Inc.**

$ 3,995.*

Dealer Inquiries Invited.

The Cromemco™ Specialists

Gunn Enterprises, Inc.
5615 Richmond, Suite 150
Houston, TX 77057
713/781-6911

*Configurations for CS-0, CS-1, CS-1H, CS-3, and Z-2H vary from $3995. to $4495.

table of contents

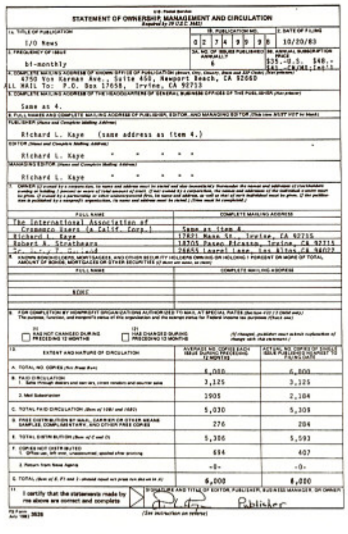
November/December, 1983 Volume Three, Number Five

I/O News ***The Official Publication of The International Association of Comemco Users is available through membership in the association. Editorial and advertising policies are designed for the enlightenment of the members in regard to new uses for, and developments of, Cromemco products and other products compatible with Cromemco systems.***

COVER FEATURES

ARTICLES & FEATURES

DEPARTMENTS



STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

I/O News — 0 2 7 4 9 9 9 8 — 10/20/83

bi-monthly — 6 — $35.-U.S. $48.-

4750 Von Karman Ave., Suite 450, Newport Beach, CA 92660
ALL MAIL To: P.O. Box 17658, Irvine, CA 92713

Same as 4.

Richard L. Kaye (same address as item 4.)

Richard L. Kaye

Richard L. Kaye

FULL NAME	COMPLETE MAILING ADDRESS
The International Association of Cromemco Users (a Calif. Corp.)	Same as item 4.
Richard L. Kaye	17821 Mase St., Irvine, CA 92715
Robert A. Strathearn	18705 Paseo Picasso, Irvine, CA 92715
[illegible]	26655 Laurel Lane, Los Altos, CA 94022

NONE

EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES	6,000	6,000
B. PAID CIRCULATION 1.	3,125	3,125
2. Mail Subscription	1905	2,184
C. TOTAL PAID CIRCULATION	5,030	5,309
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	276	284
E. TOTAL DISTRIBUTION	5,306	5,593
F. COPIES NOT DISTRIBUTED 1.	694	407
2. Return from News Agents	-0-	-0-
G. TOTAL	6,000	6,000

I certify that the statements made by me above are correct and complete — Publisher

I/O News (ISSN 0274-9998) is published bi-monthly by The International Association of Cromemco Users (a California corporation), P.O. Box 17658, Irvine, CA 92713. General offices are at 4750 Von Karman Avenue, Suite 450, Newport Beach, CA 92660. Telephone: (714) 955-0432. Controlled Circulation Postage Paid at Santa Ana, CA. POSTMASTER: Send address changes to I/O News, P.O. Box 17658, Irvine, CA 92713.

Subscriptions to I/O News are entered with membership in The IACU. Yearly memberships may be purchased for $35 (U.S. delivery address), $41 (delivery address in Canada or Mexico), and $48 (other international delivery address). Contact IACU for multi-year membership rates. Back issues of I/O News are available for $7.50 per issue. Please note: all prices are in U.S. dollars.

Return postage must be included with all manuscripts and photos submitted if they are to be returned. The IACU and I/O News accept no responsibility for the return of unsolicited materials.

Richard Kaye
Editor and Publisher

Printing
The Dot Generator

Typography
Dynacomp

Art Direction
Howard Millman Productions

input...

Editor:

I wonder whether you know of any firm making a utility program, or group of programs, for converting diskette formats from/to CDOS 5.25" DS/DD format. I know that Osborne (R.I.P.), Kaypro, Zorba and some other firms have materials of this sort, but I do not know whether any of these formats include Cromemco. Perhaps other members have encountered such a utility and could help me. Thank you for publishing my request.

Dr. Leo Greenberg
Raanana, Israel

Editor:

I thought you might be interested to know that my book on Structured BASIC will be published by MacMillan in January. It will be available in bookstores shortly thereafter. The book is titled **An Introduction to Structured BASIC for the Cromemco C-10.** The book extends an earlier version that we published by adding several new examples (file manipulation, plotting, and others). The new book is presented in revised format, with exercises added to each chapter.

On another subject, I haven't seen much in the way of news about PL/1 in I/O News, so I will provide some information. We have been using Digital Research's PL/1-80 (version 1.4) for some time with CDOS, and recently with CROMIX. It is a really good implementation of PL/1 (subset G) which works well with both Cromemco operating systems. CROMIX gives both compile and link times a real shot in the arm. In CDOS, moderate sized programs require extensive use of files by the link program, LINK-80; meanwhile, in CROMIX, the link program seldom uses files, thereby reducing link time. We have encountered only one, minor incompatability with CDOS. Upon exiting from a program, an illegal call message is issued by CDOS, but it can be ignored.

We would like to hear from other PL/1-80 enthusiasts, and can be reached by mail at 1857 Apple Tree Lane, Mt. View, CA 94040, or by phone at (415) 969-4233.

Sincerely,
Wayne T. Watson
The Software Hill

Editor:

Here's something that may be of interest to other readers. I had great difficulty in finding a BSTAM communications pprogram for the C-10, and found the only program configured for the C-10 was ProCALL, which, in my particular circumstances, I couldn't utilize.

After much aggravation, I convinced Lifeboat Associates in New York City to configure the BSTAM for Cromemco. Anyone wishing the program, which I believe costs $160 or thereabouts, should call Bob Hinkley—and **only** Bob Hinkley—at Lifeboat in New York (212) 860-0300. One nice feature with BSTAM is that it is configured to work with many other micros, so files may be easily sent back and forth.

Sincerely,
Derek Drew
New York, NY

Editor:

I am running Super Calc 2 (version 1.0, CP/M format) on a Cromemco System Three under CDOS 2.56. I run barefoot (i.e. without a CP/M simulator). All slash (/) commands operate, but a subset of the slash commands is inoperative.

Further, I cannot call up a disk directory while in Super Calc, but must quit and go into the CDOS DIR command to view a directory. This is, of course, a bit of a nuisance.

Sorcim is aware of this problem, but they state that a fix for it is low on their priority list. Additional contacts through my software dealer indicate that no fix is immediately forthcoming.

Perhaps some other IACU member has also experienced this and has already solved the problem. If so, I would be most grateful to him or her for a fix.

Yours truly,
George Collier, Jr.
1024 University
Durant, OK 74701

(Editor's Note: If you send a fix to George Collier, Jr., please send us a copy. We'll try to get the answers into the right person's hands at Sorcim. Thanks. Ed.)

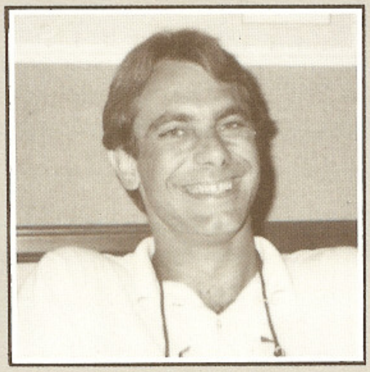
output...

Lynn Platzek

New Technical Editor

I/O News is pleased to announce the association of Bill Jaenicke as our new Technical Editor. For the past several issues, Jaenicke has been the editor of the popular **Inside CROMIX** department. His association greatly enhances our ability to deal with technical software matters, especially operating system questions.

Jaenicke has over four years experience with CROMIX (dating back to pre-release copies), and is the



Bill Jaenicke

author of the Satellite Accounting Systems SASi General Ledger and QTS Time-Keeping Systems, and his latest endeavor, I/O Archives. This last package is an archiving of all subject discussed in past issues of I/O News, and acts as a quick reference to looking up comments or articles for current use.

I/O Archives was written in dBASE II, so it requires the user to have that installed in the system in order to be able to utilize the archive commands. Jaenicke makes I/O Archives available for $25.00 per copy through SASi, P.O. Box 4251, Irvine, CA 92716. It comes with past issue data installed, as well as a user guide for keeping it up to date as new issues are received.

Many long-time readers of I/O News will immediately see the impact Jaenicke has made on the editorial content of our latest issue, and we expect continued improvement from his election to accept this much needed position.

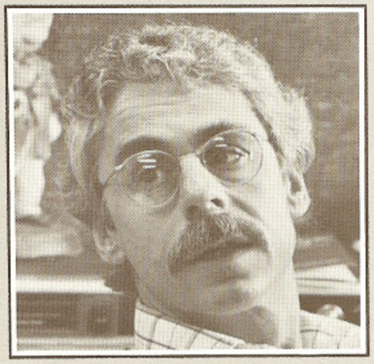
Publication Schedules

Howard Millman, the Irvine-based graphic designer who was responsible for the original design and production of I/O News, has rejoined the staff. This means more than just improving the look and readability of our issues. It means a vast improvement in our publishing schedules, an area in which we have been sorely lacking as many of you have pointed out.

Also, during the past few months we have been converting our internal processes to utilize the C-10 as our primary article preparation and editing tool. This is the first issue that has been totally processed on the C-10 and it has been the smoothest, easiest task in a long while. The C-10 has made the difference.

Trade Group Formed

John Campbell, a database researcher headquartered in Annapolis, Maryland, has formed the Cromemco Trade Association. It is a group designed to serve the interests of OEMs and other users with large numbers of Cromemco systems in use, either in one or multiple locations. We have been working quietly with Campbell for several months to help him contact such users and determine how their specialized needs can best be served. While this project is nearing the end of its formation cycle, it is still in formation and little more can be said now. If you believe that your uses fall into the general category noted, you may want to contact Campbell at: T.S.E. Research & Development, Inc., P.O. Box 2, Annapolis, MD 21404. For a brief description of the services he plans to make available, call him at (301) 263-0958.



Richard Kaye
Editor

TV Camera Interface

Continued from front cover

practically prepared through software. They may be available already in some other convenient form (a painting, a videodisc image, etc...). The ability to rapidly convert that image into a computer graphic can save time, headaches, and money.

Cromemco's SDD brings full color video digitizing down to an incredibly low cost and enhances an already extensive graphics product line.

Although functionally independent of Cromemco's SDI high resolution display interface, the SDD is fully compatible with it. The SDD's design permits it to be easily integrated into existing SDI systems, yet it has the capabilities and performance that make it suitable for a future generation of S-100 graphics products.

The SDD generates 8 bits of information for each pixel (picture element) that is digitized. This translates into 256 shades of grey or 256 colors, mapped by the SDD's inverse color map from the total pallete of 4096 colors. Digitizing red, green and blue separately produces 24 bits of data per pixel, representing an incredible 16.7 million colors. This level of performance is able to meet the demands of virtually any image processing environment.

The SDD is a slow scan type of digitizer (as opposed to a frame-grabber or flash-digitizer) and, hence, requires from 1 to 12 seconds to digitize a complete image. This speed varies based on the mode settings. There are five different SDD operating modes, all under software control, for conditions such as grey scale, inverse color mapping, and waveform analysis.

The SDD also provides extensive performance control for setting input gain, luminance, and chrominance (all under software control), as well as automatic level control (ALC), flesh tone compensation, and DC restoration of black level.

The power of video digitizing can add significant new capability to existing graphics applications and open up entirely new application areas that have not previously been feasible or cost effective.

An example of this is the concept of an image data base. Due to a sophisticated compression and storage scheme used by Cromemco's PIXSAVE program, ,an image will generally require about 10K of disk space, permitting the archiving of thousands of images on a single 20Mbyte hard disk. In an application such as parts inventory, warehouse employees may be responsible for maintaining and tracking the inventory of hundreds, if not thousands, of parts, many of them obscure or unfamiliar to the operator. A traditional computer database could provide quantity, price, backorder, and location information on a terminal screen, but an image database can provide all of this along with a digitized image of the product and a map or drawing of its location in the warehouse.

Many other applications exist for archiving images, including personnel security systems (wherein a photograph of an employee is stored along with his/her security clearance and badge number), and documentation control, where it may be important to distribute vital documents or work orders from a central location, with all remote users simply reading their up-to-the-minute version on a monitor.

The More Traditional Graphics Markets Can Also Take Advantage of the Capabilities of the SDD

The more traditional graphics markets can also take advantage of the capabilities of the SDD. Computer artists now have an entirely new medium with which to play. Artists have dreamed of the kind of full loop control that gives them the ability to take video directly into computer graphic form, and back to displayed video.

In the office, business slides can now include color photographs, or an important chart can be rapidly entered with a camera and modified with software to add data or to alter size and color.

In the factory, an SDD system could act as an inexpensive vision system, monitoring components such as bumper heights on cars or drill hole positions in sheet metal as an item passes by on the production line.

And in a laboratory, the output from an electron microscope, CAT scan, or radar device might be sent to the digitizer for the purpose of analyzing the image or to archive the collected data.

Whatever the application, a digitizer is sure to enhance or simplify computer graphics usage. Cromemco's SDD is the first full color digitizer of its kind for under $1000. It allows microcomputer graphics systems, such as the SDI, to compete with the traditional high-end dedicated graphics systems, in terms of capabilities, with undreamed of price advantages.

A final function of the SDD worth noting has, in fact, nothing to do with digitizing. The front end circuitry of the card allows the composite video input (from camera, VCR, etc.) to be split into separate red, green, blue and synchronization signals. These demodulated R, G, B, and synch levels are then passed on to the digitizing (A/D) circuitry for conversion to a digital image and to an external connector on the board. Demodulated video is thus conveniently provided in realtime by the SDD making it a video component of multiple uses. Color key-coding is also provided which allows the SDD to key, or overlay, its video output with video from another source. By taking advantage of both the keyer and demodulator capabilities of the SDD, Cromemco systems can now be employed in some of the most rapidly expanding areas of computer application, such as video training or on-the-air studio graphics, where it is desirable to overlay computer graphics directly on top of real time video.

The capability to digitize your own color video images is not only rare in a microcomputer but generally very expensive. With the low cost, high performance SDD, Cromemco once again demonstrates its innovative engineering and quality oriented product concept, bringing the highest performance to graphics systems on micros. The single board SDD is priced at just $995.

Using the C-10 "Computer" Port for Interface to External Equipment

By Leigh R. Thomas

SUMMARY

This article describes the use of the "COMPUTER" connector socket on the rear of the C-10 with particular reference to using this port for data I/O to or from peripheral devices under program control.

INTRODUCTION

Many C-10 users may be interested in connecting peripheral serial devices, like bitpad digitizers or speech generators for example, to their personal computer or possibly in writing specialized programs or drivers to handle file communication with another computer.

The C-10 user manual contains only sufficient information to enable a user to connect to a remote system using the Connect function. It does not give specialized information regarding how this port may be used or how a user's application program may interact with it.

The C-10 Technical Reference Manual is more explicit, giving port addresses but, again, no hard data about handling this port. The reader of the technical manual is referred to the UART data sheet for details. For those users who have neither the technical manual nor access to data sheets, this article describes a simple method of driving this "computer" port on the rear of the C-10. The program information is for 32K Structured BASIC, but the principles are general and may be transported to other application languages.

SERIAL COMMUNICATION

The C-10 has three "sockets" on its rear panel in addition to the keyboard connector. One of these is for disk drives, one is for printers and the third, labelled "computer," is specifically for remote connection as a terminal to another computer either directly, or via a modem link.

The printer socket is, in fact, two sockets in one; a parallel and a serial output socket. The serial section of this printer socket is limited in function, having a preset baud rate of 1200 and one-way communication only—out from the C-10 to the remote device (typically printer).

The so-called "computer" socket is general purpose. It has programmable baud rates and stop bits.

There are two ways to effect enabling of this port. In normal use the C-10 may be connected via this socket to a remote computer. If, when the C-10 boosts, it finds a disk drive attached, it ignores this port and begins interaction with the user in the normal way, displaying its menu. Should the user wish to connect to the remote system, he may enter the command "CONNECT" at the menu prompt. This instructs the C-10 to send carriage return characters out the rear computer port using the default protocol (if no specific changes were initiated—see later) and "listen" for a response. If no responding carriage returns are discovered, a new, lower baud rate is used and the sending cycle is repeated. Ultimately, if all is well, a response will be found and the message "C-10 Connected to Remote System" is displayed. At this point the C-10 is behaving only as a terminal, not as a computer, and the user may interact with the remote system in the normal way as one would on any terminal.

The CONNECT function has enabled the rear computer socket and "opened the channel." To revert to computer mode, the user may enter Control-Shift-Q. This will cause the C-10 to reboot as normal. Note though, the rear socket is still "open."

A similar procedure may be invoked from the CROS. Should a user hit the ESC key just after power-on, booting is inhibited and interaction with the resident operating system may take place. The "i" option displays a list of features pertaining to the serial communication method used at the socket and a user may instruct the system at this stage what baud rate, stop bits, number of data bits and other features are relevant to their task. These instructions jointly configure the port to the requirements of the peripheral device. Note however that the port is not "enabled" or "opened" by this procedure alone—it is simply configured appropriately.

Should the user so desire, he may CONNECT to the remote system at this stage using the CROS command "C" with results as described above. A successful connect will enable or "open" the port.

If the user is intending to communicate with a peripheral device rather than a computer, the connect command loses some of its meaning. Whilst a computer may echo carriage returns and then accept commands, devices generally will not. A user seeking to interface to peripheral devices will usually wish to do so with the C-10 in computer-mode, not terminal-mode, and control the connection port via an application program. The "connection" may be established by deliberately fooling the hardware—shorting the send and receive pins (4 and 5) together at the "computer" DB-9 connector. This will return the "I am connected" message and simultaneously enable the port but is, however, unnecessary. The application program may enable the port as described below.

HARDWARE AND PROGRAMMING CONSIDERATIONS

The "computer" socket is in fact 4 I/O ports, numbered 56, 57, 58, and 59 decimal (38H to 3BH). Port 56 is the data input/output port, port 57 is the status port, port 58 is the command port and 59 the control port.

A user program may send serial

data to a remote device by outputting this data to port 56. For example, in 32K BASIC, a data output command may look like:

```
Ch$ = "R"
Out 56,Asc(Ch$)
```

to send the letter "R" to the device.

An application program may obtain serial data from the port in a similar way. For example:

```
Indata = Inp(56)
Ch$ = Chr$(Indata)
```

will fetch a single character from the computer port 56. Repeating this input function may, however, repeatedly read the same data. The status port, 57, contains information regarding whether a new character has arrived which has not yet been read in. If we name the 8 bits of the port as:

```
MSB                                      LSB
bit7 bit6 bit5 bit4 bit3 bit2 bit1 bit0
```

then bit3 of port 57 contains this new-character information. That bit is normally a zero but changes to a one when a new character arrives from the remote device. It will remain a one until the application program running in the C-10 fetches the character data from port 56 via an INPut instruction, whereupon the bit reverts to a zero. Thus an application program will typically contain a "polling" line like:

```
1000   If Binand(Inp(57),8) = 0
           then 1000
```

This line will execute indefinitely in a tight loop, continuously checking bit3 of port 57, until a new character arrives. The "8" is binary 00001000, and thus the binary AND mask functions as required.

The application program interfacing with external equipment cannot function until the port is enabled. If "connect" has not been used, the program must explicitly enable the port before proceeding. This needs only to be performed once at the start of the program, not each time a character is fetched. This enabling is performed by setting the least significant bit, bit0, of the command port, 58, to a one. Thus, a program line like:

```
Out 58,Binor(Inp(58),1)
```

is used. The "1" is binary 00000001 and the binary OR function coupled with the INPut function allows the port to be read, its LSB set and then the data output again, leaving the other bits intact.

A NOTE ON SPEED

In these examples, 32K Structured BASIC is used to control the port. When fetching data from the port, no automatic input buffering is available in the C-10. The application program must handle this. It is therefore critical that characters do not arrive so fast that the program cannot "keep up." 32K BASIC, whilst able to function at respectable speeds, cannot match a machine-language routine for speed in cases like this, where several program steps must be processed for each new character. I have found that 32K BASIC in a tight, well-structured routine, will happily function at 300 and 600 baud and may even be persuaded to go faster.

Depending upon the requirements and capabilities of the remote device, the application program may keep count of characters arriving and send XON, XOFF (or similar) instructions as required to establish the communication protocol.

About the Author

Leigh R. Thomas works with several CROMIX systems as a Biomedical Engineer, and plays with a C-10 at home. He can be reached by mail at:

1/127 Burns Bay Road,
LANE COVE
N.S.W. 2066 Australia

Please forward all suggestions and/or reservations for his forthcoming book directly to him.

Using dBASE II to Set the System Clock

by Jim Gunkel

Recently, I reviewed the beta test version for a new release of dBASE II. The new features will make it even more of a pleasure to do data base management system type work. I have worked with data base management systems (DBMS) for the past 15 years using languages ranging from Assembly and COBOL, to the new program generators, such as Cromemco's DBMS and DBR. The latest DBMS type languages for microcomputers have taken the needed features from the past and applied some lessons learned to produce significantly improved releases that are much easier to program. This is not meant to be a discussion on the theory of DBMS, or the pros and cons of the different types. Many college level courses and seminars are available for that. It is meant to be a real world discussion of some of the power and flexibility of the new DBMS versions such as dBASE II.

Many words have been written on the power of dBASE II. I agree that it is a very powerful programming language and the ease of programming with it is a pleasure when compared to older methods available to the microcomputer user (such as BASIC). However, no matter how powerful or flexible any language may be, it is not going to gain wide acceptance unless it can do some "neat" things. These neat additions, more commonly called "bells and whistles," are not necessary, but can make the difference with a user or customer who is not interested in the inner workings of the computer. Most want a system that works and requires minimal effort or training. What a programmer may deem as "neat but not needed," may be the very item that the individual sitting at the terminal would consider very worthwhile. It should not be a surprise to anyone that dBASE II has become very popular. This is due in part to its use of a more natural "sentence structure" and its ability to do many items conveniently that "tailor" the system to your particular system. This may be sending special codes to the printer to change to double-wide print and back, or changing the CRT terminal to reverse video for prompting during user input sessions. The purpose of this article is twofold. First, to provide a useful utility module for dBASE II programs. Second, using this same example, to give an indication of the power available to dBASE II programmers for Cromemco Users (both CDOS and CROMIX) of dBASE II.

dBASE II allows the use of PEEK and POKE commands. These commands open the door to many features using CDOS calls (SIM.BIN in CROMIX). One such set of calls are those for setting or reading the date and time. If you have a Cromemco 3101 or 3102 terminal with a clock function, this routine will allow you to read or change the time. Using the read portion of this function, the computer can time routines (sorts, indexing, etc.) by saving the time as a memory variable for later retrieval or analysis.

These programs are intended to be called by the dBASE "DO" command. There are many different programming techniques or styles that can be used. The routines are presented as separate modules to show new or inexperienced dBASE II users their independent capabilities. In actual use, they can be combined or further divided as particular needs require. For instance, a hard disk system minimizes the time delay of extra read activities of separate modules. With a floppy disk based system, disk access being much slower, these extra reads may indeed become a factor. The extra floppy disk wear should also be considered.

Five "module" listings are included and are arranged as follows:

SAMPLE.CMD is your main program which could be a menu, etc. In this case, it is simply a lead program to start the sequence and then display the results. I suggest that you do the following to fully demonstrate these routines: Reset the system (if necessary turn it off—then back on) and boot in CDOS or CROMIX. Using CDOS as an example—use the STAT utility and do a STAT/A. DO NOT DO A STAT/DT. When STAT comes up it will not have a date or time displayed. Using dBASE II do the following: DBASE SAMPLE. The program will mandatorily ask for the date and time since it determined that they had not yet been set. After the dBASE II prompt comes back (a period), type in QUIT and do a STAT/A again. This time you will notice that both the date and time have been set. Now, if you do DBASE SAMPLE again WITHOUT a reset, you will find that you are given a choice on setting the date. This is a lot of work to simply set the time, but lots of potential is there when using this as a part of a larger program. You can read the time at the end of the session (or part of a program) and read the time at the end of the session and compute the time on the system. It is useful for dating reports that the system may generate by doing a read as part of the report program just before printout.

UTWRDATE.CMD stands for UTility WRite DATE.CoMmandD program and is the program that will determine if your system clock is already set. It will ask you a series of questions depending on the status of your system. If necessary, it will then set the date and time functions in CDOS or CROMIX.

UTRDDATE.CMD is the UTility ReaD DATE.CoMmanD program to read both date and time without any change to the system clock (no write activity).

UTMONTH.CMD is a utility program to convert month information from numeric data into a fully-spelled month and its three letter abbreviation.

UTRDTIME.CMD is the UTility ReaD TIME.CoMmanD program to read time only in applications that don't need the date checked (no write activity).

See program ▸

```
* "SAMPLE.CMD = Sample Main Program
* "20   Jan 1983   COM SOFT"
* "used to get the current month and day and set date function"

SET talk OFF
ERASE

DO utwrdate

STORE (mm-1) * 30 TO dd:sum
STORE 31 TO dd:max
STORE "ERROR" TO month
DO utmonth

STORE str(dd,2) + " " + month + " " + str(year,4) TO asofdate
STORE mon + " " + str(yy,2) TO monyr
STORE dd:sum + dd to daysgone
ERASE

@  4, 5 SAY "You have now set the system clock"
@  8,10 SAY "Date is set to                  " + asofdate
@ 12,10 SAY "Date - abbreviated              " + monyr
@ 16,10 SAY "Days already gone this year " + str(daysgone,3)

SET talk ON



* "UTWRDATE.CMD = UTility WRite DATE.CoMmanD"
* "21   Sep 1982   COM SOFT - Jim Gunkel"
* "used to get the current date and time in operating system"

ERASE
@  2, 5 SAY "Getting the -DATE-"
@ 10, 5 SAY "Getting the -TIME-"

DO utrddate

STORE f TO setdate
IF mm = 0 .OR. hh = 0
  IF FILE ('pastdate.mem')
    RESTORE FROM pastdate
  ENDIF
  STORE t TO setdate
  STORE "Y" TO action
ENDIF

STORE 42240 TO freeroom
STORE "STRING" TO string
STORE 41000 TO i
SET CALL TO i+5

@  4,10 SAY "Current Year    [yy]  " GET yy
@  6,18 SAY "Month    [mm]  " GET mm
@  8,18 SAY "Day      [dd]  " GET dd
@ 12,10 SAY "Current Hour    [hh]  " GET hh
@ 14,18 SAY "Minute   [mm]  " GET mn
@ 16,18 SAY "Second   [ss]  " GET ss

IF setdate
  @ 18, 5 SAY "Please Correct the Date / Time"
ELSE
  @ 18, 5 SAY "Do you want to CHANGE the Date / Time   [N]   "
  SET console OFF
  WAIT TO action
  STORE !(action) TO action
  SET console ON
  ?? action
ENDIF

IF !(action) = "Y" .OR. setdate
  READ
  POKE i+5,237,91,0,165,237,75,2,165,65,14,145,205,5,0,201
  *     ld de,(a500)    - [42240]        get time
  *     ld bc,(a502)    - [42242]        ssmnhh format
  *     ld b,c
  *     ld c,91         - [145]          write time
  *     call 5
  *     ret
```

Continued

```
  POKE freeroom ,hh
  POKE freeroom + 1,mn

  POKE freeroom + 2,ss
  CALL string

  @ 20, 5 SAY "Setting the Date / Time in your system"

  POKE i+5,237,91,0,165,237,75,2,165,65,14,143,205,5,0,201
  *      ld de,(a500)    - [42240]       set date
  *      ld bc,(a502)    - [42242]       ddmmyy format
  *      ld b,c
  *      ld c,8f         - [143]         write date
  *      call 5
  *      ret
  POKE freeroom ,yy
  POKE freeroom + 1,mm
  POKE freeroom + 2,dd
  CALL string
ELSE
  CLEAR GETS
ENDIF

STORE yy + 1900 TO year
RELEASE action,freeroom,i,setdate,string
SAVE TO pastdate

RETURN
```

```
* "UTRDDATE.CMD = UTitity ReaD DATE.CoMmanD"
* "21   Sep 1982    COM SOFT"
* "used to read the current date and time"

STORE 42240 TO freeroom
STORE "STRING" TO string
STORE 41000 TO i

SET CALL TO i+5

POKE i+5,14,144,205,5,0,237,67,0,165,50,2,165,201
*       ld      c,90            - [144]         read date
*       call    5
*       ld      (a500),bc       - [42240]       save date at a500h
*       ld      (a502),a        - [42242]       yymmdd format
*       ret
CALL string

STORE PEEK(freeroom   ) TO yy
STORE PEEK(freeroom+1 ) TO mm
STORE PEEK(freeroom+2 ) TO dd
STORE yy + 1900 TO year


POKE i+5,14,146,205,5,0,237,67,0,165,50,2,165,201
*       ld      c,92h           - [146]         read time
*       call    5
*       ld      (a500),bc       - [42240]       save time at a500h
*       ld      (a502),a        - [42242]       hhmmss format
*       ret
CALL string

STORE PEEK(freeroom   ) TO hh
STORE PEEK(freeroom+1 ) TO mn
STORE PEEK(freeroom+2 ) TO ss

RELEASE freeroom,i,string

RETURN
```

```
* "UTMONTH.CMD = UTility MONTH.
  CoMmanD
* "21    Sep 1982    COM SOFT"
* "used to setup names of months"

DO CASE
   CASE mm = 1
      STORE "January" TO month
   CASE mm = 2
      STORE 28 TO dd:max
      STORE dd:sum + 1 TO dd:sum
      STORE "February" TO month
   CASE mm = 3
      STORE dd:sum -1 TO dd:sum
      STORE "March" TO month
   CASE mm = 4
      STORE 30 TO dd:max
      STORE "April" TO month
   CASE mm = 5
      STORE "May" TO month
   CASE mm = 6
      STORE dd:sum + 1 TO dd:sum
      STORE 30 TO dd:max
      STORE "June" TO month
   CASE mm = 7
      STORE 181 TO dd:sum
      STORE "July" TO month
   CASE mm = 8
      STORE 212 TO dd:sum
      STORE "August" TO month
   CASE mm = 9
      STORE 243 TO dd:sum
      STORE 30 TO dd:max
      STORE "September" TO month
   CASE mm = 10
      STORE 273 TO dd:sum
      STORE "October" TO month
   CASE mm = 11
      STORE 304 TO dd:sum
      STORE 30 TO dd:max
      STORE "November" TO month
   CASE mm = 12
      STORE 334 TO dd:sum
      STORE "December" TO month
ENDCASE

STORE $(month,1,3) TO mon
RETURN
```

```
* "UTRDTIME.CMD = UTility ReaD TIME.CoMmanD"
* "21    Sep 1982    COM SOFT"
* "used to read the current time"

STORE 42240 TO freeroom
STORE "STRING" TO string
STORE 41000 TO i

SET CALL TO i+5

POKE i+5,14,146,205,5,0,237,67,0,165,50,2,165,201
*       ld c,92h          - [146]         read time
*       call 5
*       ld (a500),bc      - [42240]       save time at a500h
*       ld (a502),a       - [42242]       hhmmss format
*       ret
CALL string

STORE PEEK(freeroom   ) TO hh
STORE PEEK(freeroom+1 ) TO mn
STORE PEEK(freeroom+2 ) TO ss

RELEASE freeroom,i,string

RETURN
```

About the Author

James R. (Jim) Gunkel was, until mid-1983 when he retired, Program Manager for the United States Air Force working with simulation devices for the military.

During his air force career, Col. Gunkel became a computer software consultant specializing in CP/M and dBASE II. He has contributed several articles to I/O News, recapped below:

Vol. I, No. 3 — Tips on CDOS Calls

Vol. I, No. 5 — Mapping Routines for Floppy Disk Clusters

Vol. II, No. 4 — Putting Your Data Files in Order

Gunkel has moved much of his programming efforts to dBASE II due to its flexibility and power, and recently received a BETA copy of dBASE II.

Gunkel has available several utility routines using dBASE II commands, in addition to the subjects of this article. All have proven very popular with his clients. Since his retirement from the air force, he has opened a private software consulting firm and reports that he is enjoying every minute of his new life. Gunkel can be reached at:

COM SOFT
127 West Erie Drive
Tempe, AZ 85282
Phone: (602) 967-6698

CROMEMCO 1984

Continued from front cover

sorting-out process. The first part of the question is not one that we could easily answer, so we arranged an interview with **Dr. Harry Garland, President of Cromemco, Inc.** The entire interview follows.

I/O: Dr. Garland, when The Cromemco Story first appeared Cromemco was less than five years old. At the end of that story we made a prediction that "The future will be very exciting." By the time this interview appears, Cromemco will be about eight years old. Has the intervening period been exciting?

HG: It certainly has. As technology has advanced over the past few years we have had an opportunity not only to keep our product line current but to lead in the application of new technology in microcomputer systems. As you recall three years ago our product line consisted exclusively of Z-80 based, S-100 systems. Now we offer the advanced 68000 Microprocessor with our bus-oriented systems and have expanded our Z-80 based offerings to include the C-10 Personal Computer. I feel particularly proud that we were able to make this transition smoothly and in a way that has allowed our early users to upgrade their systems to the newer technology.

"...Cromemco started doing business in 1975, which makes us...the oldest company in the microcomputer system business."

I/O: We have read in the newspapers and the trade press that there is a "shakeout" occurring in the computer industry. How is Cromemco doing?

HG: I am pleased to report that we are doing extremely well. As you know Cromemco started doing business in 1975 which makes us today the oldest company in the microcomputer system business. For eight years now Cromemco has experienced sales growth every single year, and has been profitable every year. I believe that this track record would be the envy of any company in the industry.

In general, I think when the press has talked about a shakeout in the computer industry in many cases they have extrapolated from activities in the home computer market, which is quite remote from where Cromemco products are marketed, and in addition have extrapolated from the predictable failures of some very young and poorly managed companies in our own industry. In so doing they have come up with generalizations which in many cases are just plain inaccurate.

In short, Cromemco has had an eight year track record of profitability and is entering 1984 with the strongest financial position in the company's history.

I/O: You mentioned earlier that you decided to go with the 68000 microprocessor for your high end systems. Why did you choose the 68000?

HG: Well, that was a very important decision for us. Two criteria were used in making the decision. First we wanted to offer the most capable of the new generation microprocessors for our customers and secondly we wanted a migration path in the future so that software written for this new microprocessor could be upgraded to increasingly more capable machines. The 68000 was clearly the winner in the first category in that it is the only one of the new generation of microprocessors with a full 32 bit internal architecture and it is the only one of the new processors which supports a full unsegmented 16 megabyte address space. In addition when Motorola announced the 68000 they also announced their plans for a follow-on family of products including the 68010 and the 68020 microprocessor. It was important to us that software written for the 68000 could run on these future processors with little or no modification. By adopting the 68000 then we were able to offer our customers the most capable of the new generation microprocessors and in the future provide a migration path for our customers

which would allow them to use increasingly more capable microprocessors without having to rewrite their software.

I/O: When will Cromemco introduce products with the 68010 and the 68020 microprocessors?

HG: As you know we currently offer our dual processor unit (DPU) with a 68000 microprocessor and a Z-80 microprocessor on it. We are currently working on a new dual processor board which will include the 68010 with the Z-80 processor. This board is planned for introduction by the end of 1984. As for the 68020, this microprocessor is still in the design stage at Motorola and given the complexity of the microprocessor and the design challenge it presents, I would be surprised to see it included in systems from Cromemco or any other manufacturer until late in 1985 or perhaps in 1986.

I/O: Will there be any additions to

the C-10 product line during 1984?

HG: Yes indeed. Very soon for example we will be introducing an eight inch floppy disk drive for the C-10 which we call our Model CLD. The C-10 is designed to support either 5-inch or 8-inch floppy drives and in fact can support any mixture of up to four 5-inch and 8-inch floppies. Due to the higher storage capacity and higher speed of the 8-inch floppy drive I believe the CLD will help to position the C-10 into many new markets. In addition, we plan to introduce refinements to the C-10 operating system and software to increase its speed and to increase its compatability with a wider range of CP/M software.

I/O: Some users have told us that the Cromemco 20 megabyte disk drive is slower than the earlier 11 megabyte disk drive that you have now discontinued. Why is that?

HG: You have to remember that the 11 megabyte disk drive used voice coil technology rather than stepper motor technology as is used in our current 20 megabyte drive. Also the 11 megabyte disk drive had a price which was nearly twice that of our 20 megabyte drive even though it had little more than half the storage capacity. The 20 megabyte drive now used in all our H-series systems has been an extremely reliable drive, and that is important to us. It is true that there are faster disk drives on the market, but they tend to be considerably more expensive than our 20 megabyte drive.

"The Bunker Ramo report...concluded that the Cromemco systems best met the...requirements of the Air Force."

I/O: Will you be adding a faster hard disk drive to your product line?

HG: Yes. We have recently introduced, for example, our SMDI controller and SMD driver software package. SMD stands for Storage Module Drive and with our new SMDI interface, Cromemco users now have access to a very large selection of high speed and high capacity SMD disk drives. In addition, we are evaluating a number of high speed 5-inch drives using the so-called ST-506 interface and we plan to introduce such a drive with a new Cromemco ST-506 interface card during 1984.

I/O: In looking at new subscriptions as they come in to I/O News we have noticed a large number from U.S. Air Force Bases around the world. Why is that?

HG: Well it is very interesting. It turns out that 3 years ago the United States Air Force commissioned Bunker Ramo to do a study to recommend which would be the most appropriate microcomputer for a variety of U.S. Air Force applications. The Bunker Ramo report, which is an unclassified public document, identified 149 possible microcomputer systems which might be appropriate to U.S. Air Force needs. Of these 149 the report concluded that the Cromemco systems best met the strict requirements of the Air Force. Virginia Communications Associates, a Cromemco OEM, subsequently won a competitive bid for supplying Cromemco microcomputer systems to the U.S. Air Force. Today our systems are used at U.S. Air Force bases around the world and Virginia Communications Associates has done an outstanding job of providing the requisite system integration, support, and training.

I/O: Do you plan any major new product announcements in 1984 which you have not yet discussed in this interview?

HG: Yes.

I/O: Fair enough! I am sure 1984 will indeed be another exciting year for Cromemco.

Editor's note:

We will try to update "The Cromemco Story" via interviews with Dr. Garland on an annual basis. To that end, we ask you to send us your questions as to the directions the microcomputer world—and Cromemco, in particular—may be heading. We will compile these and use them in our next interview.

Dr. Harry Garland, President and Co-founder of Cromemco, surrounded by C-10s in final testing stage.

The Menu Generator

by Bill Jaenicke

As a special New Year's treat, we're happy to be able to provide you with something that many have requested, a software review. For this issue, Bill Jaenicke, our TecEd, tried out MenuWare Inc.'s latest endeavor: The Menu Generator.

Product: The Menu Generator
Producer: MenuWare, Inc., 16351 Skyline Boulevard, Woodside, CA 94062. (415) 851-3400
List Price: $249.00

If you've ever worked with the CROMIX Operating System, you know what it's like. Very powerful, but not very friendly. The Menu Generator can change that. Basically, it enables you to "menu-ize" your entire set of computer operations. All of your various applications can be integrated into a user-friendly menu environment.

The Package

The Menu Generator package consists of a handsome black binder, embossed with the MenuWare logo, and containing the system disk, installation instructions, documentation, software registration form, and software license agreement. The registration form and software license agreement are fairly standard, except for a clause pertaining to maintaining backup copies: none can be made. The system disk contains two copies, each of which can be installed once. Thereafter, you send the disk back to MenuWare for re-initialization and update, and they'll send it back to you. It's a novel, but reasonable, approach to limiting software piracy.

Product Description

The need addressed by The Menu Generator is one which is omnipresent among CROMIX end-users: what can be done to facilitate quick and effective use of the computer by inexperienced people. The solution is simple: make it all menu-driven. The Menu Generator provides this capability. Custom menus can be designed to meet any set of needs. By designing an appropriate hierarchy of menus, users of the system need never interact directly with the operating system.

Having the users interface the system by way of menus is also a simple way to limit their activities. If there isn't a menu option for it, they can't do it. You can do away with the ownership and access problems inherent to a CROMIX system with privileged and nonprivileged users: everyone can be privileged, yet controlled.

Documentation

The manual provided with The Menu Generator is short (39 pages) but complete. The style is clear and concise, the content without fat. It provides examples of the various screen displays and takes you through sample situations.

It is functionally organized, beginning with important definitions (of which there are but a few). The technical features are detailed, including various size limitations. What limitations are present are hardly restrictive. The remainder of the manual deals with the procedures involved in the design, modification, and implementation of various menu schemes. Included at the end are the patch locations which control terminal features.

In addition to the written documentation, extensive online 'help' documentation is provided. Between the written documentation, the online help, and the example menus provided, I found there to be ample information to quickly and easily become proficient at creating custom application menus.

Installation

The installation procedure is straightforward. The instructions are the first thing you see when you open the binder. You simply mount the system disk, set the directory to the mounted disk, and type "install." The automated installation routine creates a '/usr/pkg/gen' directory into which the majority of the system is copied. Other programs are copied into the '/cmd' and '/bin' directories. You are given an option of installing the online help documentation. The entire package (help included) only takes up about 155K.

Operation

Once installed, the Menu Generator can be run by typing 'genny' at the CROMIX command line. What you get is a menu (what else?). From this menu you can work with 1) Menus; 2) Questionnaires; 3) Training Programs; 4) Adventure Games; 5) Edit a Command String Library; or 6) Print a Command-String Library.

Menus consist of pages. Each page can have a heading, up to nine selectable line items (options), and an optional, two-line message. The system date and time are displayed on the top line. Each selectable option can be either a command or a page. Commands consist of 'Command-Strings,' which are sequences of CROMIX commands similar in every detail to the usual CROMIX command file. In fact, a valid command-string might simply call an existing CROMIX command file. Pages lead to another set of options, which in turn

can lead to other commands or pages.

Menus can be created, copied, deleted, edited, printed (on screen or spooled to the printer), and statistically analyzed. Excellent facilities are provided to accomplish this with an absolute minimum of time and effort.

Questionnaires, Training Programs, and Adventure games are simply variations of the Menu theme. Each group has its own directory and startup routine. Questionnaires can be designed to get information from a user and execute a command-string as a result. Training programs can be designed to assist in learning a subject in a multiple choice format. An Adventure Games format is provided for amusement and imagination expansion.

Limitations

The size of the displayed portion of a menu is limited to around 30,000 bytes. The command-string portion is limited to about a megabyte. A menu can consist of more than 1,000 pages, allowing 1,000 plus choices, with up to 1,000 commands. Command-strings can be up to 512 bytes in length. I don't see any trouble here.

The only other limitations are those imposed by your own imagination. With a little thought, and a working knowledge of CROMIX, complete systems of menus can be designed to accommodate any unique requirements. Unless you are familiar with CROMIX, you will probably want to consult with someone (MenuWare dealer) to assist in the configuration of your menu system.

Operating Features

The first thing that impressed me with the operation of The Menu Generator was it's speed. New menus are displayed almost instantaneously after pressing the key. It is really fast! This is a result of efficient programming in 'C'.

I was also impressed by the neatness of the screen displays throughout. The CRT functions, such as underlining and half-intensity video, are used with taste, and result in pleasant, balanced, and easy to read displays.

And it is very simple to learn and use. The editing routines are standardized, and there is always online help available. Very complete verification of user input prevents "blowouts" by accidentally (or intentionally) giving a nonsense response. Once the menus have been created, it is a simple matter to run them. Users can be set up, via the CROMIX 'passwd' utility, such that they start at a 'home' directory and execute their own menus. When they exit the menu they are logged off the computer. No fuss, no muss, no danger of someone accidentially initializing the hard disk.

Special Features

MenuWare went all out in making The Menu Generator easy to use. For example, when creating a menu, and specifying a command-string, you have the option of entering the command-string directly or reading it from a command-string library. So you can build these libraries and then construct custom menus by simply selecting labelled parts.

A drawback common to many menu-driven systems is that you must follow a specific path of selections to reach a desired function, and then reverse the path to get back. The Menu Generator eliminates this nuisance by allowing random page access. When editing or using a menu it is possible to go to any specific page (page numbers are displayed as part of the menu) by entering the page number. This way you can jump to any menu from any menu. A very nice feature!

While creating or editing menus you also have the ability to 'fork a shell.' This useful facility makes it possible to jump out of what you are doing, get the CROMIX prompt, and execute anything you want through CROMIX. When you EXIT, you are returned to where you left off when you forked the shell. For example, you might want to edit a command-string library or check on a file name, during the course of editing a menu.

Another useful feature is the 'getargs' command. This command enables menus to be interactive with the user. It works similarly to argument substitution in CROMIX by enabling a command-string to prompt for and retrieve information from the user. This would be used, for example, in a command-string for copying a file: the user would be prompted for the file to copy, and where to copy it to.

Applications

It was soon apparent that there was much more here than met the eye. The possible applications are innumerable. The benefits to be derived are inestimable. Precise, tailor-fitted, menu-driven systems can be quickly and easily generated. Users need never be bothered with learning the operating system commands. From Login to finish, it's menus all the way.

With a little imagination, and know-how, there's no limit to what can be done. It is so flexible that I'm sure that people will continue to come up with novel applications for a long time to come. A definite "must have" for any CROMIX-based system.

Areas for Improvement

All in all, there really isn't anything to be particularly critical about. The documentation regarding 'getargs' could be expanded upon and clarified. Additional examples would help with this. As far as the operations are concerned, my only observation was that there was not a way to change a 'page' line item to a 'command' line item, and vice-versa. I spoke with Terry Bogart, President of MenuWare, Inc., and was informed that others had brought this up and that they were working on an update that would take care of it. As it stands now, I feel that The Menu Generator is a good example of quality software.

Summary Evaluation

Ease of installation	Excellent
Ease of use	Excellent
Documentation	Very good
Speed	Excellent
Reliability*	Excellent
Efectiveness*	Excellent

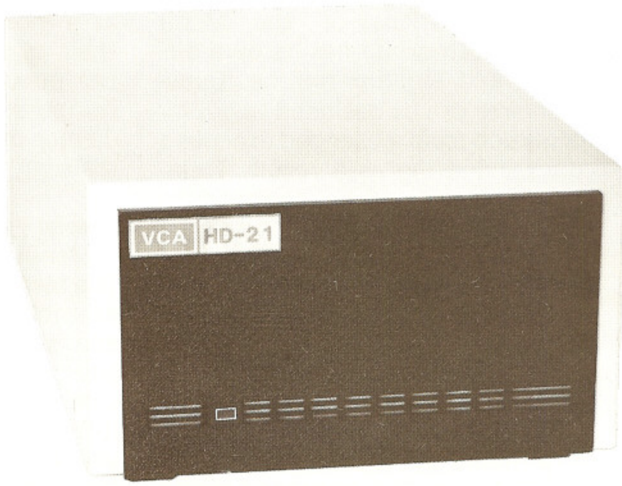
*Reliability = No bugs encountered
*Effectiveness = Does it provide benefits it was designed to

EZCROMIX

EZCROMIX is a user-friendly front-end process to CROMIX. It allows the most inexperienced user to quickly and easily become productive on a CROMIX computer system. At the same time the more sophisticated user can easily bypass EZ CROMIX to immediately get to the full power of CROMIX.

The experienced user (system administrator) can use the data base management part of this package to set up any menus tailored to specific needs of any organization.

HD-21



SOFTWARE

- Communications
- Micro Pro Products
- Microsoft Basic, Complier & Interpreter

TRAINING: HARDWARE MAINTENANCE

CROMIX

SYSTEM PERIPHERALS

- OKIDATA Printers
- NEC Printers
- RACAL-VADIC Modems
- GTCO Digitizers
- ZENITH Z-29 Terminals
- HOUSTON INSTRUMENTS Plotters
- RGB Monitors 13 "AND 19"

VCA CORPORATION

Call or Write Today

501 Church Street, N.E., Suite 200
Vienna, Virginia 22180

Telephone: (703) 281-4666
Telex: 469088

Local Cromemco User's Groups

Arizona Association of Cromemco Users
Contact: Jo Ann Drake, President
2207 West Eugie Avenue
Phoenix, AZ 85029
(602) 993-9589

Bay Area Cromemco Users & Programmers (BACUP)
Contact: Raymond Barglow or Alan Walworth
United Word & Data Processing
2345 Fulton Street
Berkeley, CA 94704
(415) 841-0708 or (415) 548-2692

Cromemcohorts
Contact: Dr. Brent Lowensohn
4747 Sunset Blvd.
Los Angeles, CA 90027
(213) 667-8972

Cromemco Users' Group Holland (CUGH)
Contact: Joop Kohler, Secretary
P.O. Box 120
2910 AC Nieuwerkerk a/d IJssel
The Netherlands 01803 - 3300

Greater Detroit Area Users' Group
Contact: Frank D. Baber
P.O. Box 909
Warren, MI 48090
(313) 575-4607 or 759-2152

Cromemco Users' Group
Contact: Peter Norman
The University of Newcastle Upon Tyne
Department of Chemical Engineering
Merz Court, Claremont Road
Newcastle Upon Tyne NE1 7RU
England

Newcastle 28511, Ext. 3278
*Publishes Cromemco Users' Newsletter (CUG)

Insystems Pty. Ltd.*
Contact: Norman Rosenbaum
337 Moray Street
South Melbourne, Victoria
3205 Australia
(03) 690-2899, telex AA30458
*Publishes "Cromemco UPDATE
a bi-monthly newsletter

Illinois Users' Group
Contact: Jim Knowles
P.O. Box 631
Elgin, IL 60120
(312) 695-7775

Indonesian Cromemco User's Group (ICUG)*
Contact: Zafir M.A. Pontoh
Computation Lab
Department of Regional &
City Planning
Bandung Institute of Technology
10 Ganesha
Bandung, Indonesia
(022) 82051 ext. 360
*Publishes "BERKALA ICUG,"
a monthly newsletter

Microcomputer Users' Group
Contact: Jim Lenz
1165 Barbara Drive
Cherry Hill, NJ 08003
(609) 428-6701

Netherlands Users' Group
Contact: R. van Wezel
Deurloostraat 115 hs.
1075 HX Amsterdam
The Netherlands
(020) 761 549

Northwest Association of Cromemco Users (NWACU)
Contact: Jim Illman
403 S. Brandon
Seattle, WA 98108
(206) 763-2099

North Texas Cromemco Commercial Users' Group
Contact: Jerrell Johnson
1131 Winterwood
Lewisville, TX 75067
(214) 221-1437
Or call Rocky Hall
@ (214) 398-1595
Meets first Thursday each month

NY, NY Users' Group
Contact: Charles Perrella
45F Route 303
Valley Cottage, NY 10989
(914) 268-5137

SaCromemco Users
Contact: Alan Whitman
Box 244
Rancho Cordova, CA 95670
(916) 635-6070

bits & bytes, nibbles & tweaks

Patches to Screen for the H-19 Video Terminal

Sam Guccione of the Terry Campus of Delaware Technical and Community College writes that he has developed patches for SCREEN which allow the H-19 to function correctly in all commands except for the blinking used to highlight markers set, etc. In that instance, he used the reverse video mode in place of the blinking. The patches are included in two tables, one for CDOS, the other for CROMIX. He cautions that the H-19 must be set to the Heath mode (i.e., switch five of S402 must be in the off or zero position).

Guccione concludes, "After using SCREEN with these patches, we wonder how we got along without them. The instructors who use our Cromemco System Two think it is just great."

TABLE I — Patches to SCREEN version 1.24 for CDOS

Memory Location	Before	After
0CB	46	59
0CC	00	00
0CD	1B	00
0CE	5A	00
0CF	00	00
0D0	1B	00
0D1	5A	00
0D2	00	00
0D3	1B	1B
0D4	6C	70
0D5	00	00
0D6	1B	1B
0D7	6D	71

TABLE II — Patches to SCREEN version 1.35 for CROMIX

Memory Location	Before	After
0CD	46	59
0CE	00	00
0CF	1B	00
0D0	5A	00
0D1	00	00
0D2	1B	00
0D3	5A	00
0D4	00	00
0D5	1B	1B
0D6	6C	70
0D7	00	00
0D8	1B	1B
0D9	6D	71

Cromemcohorts...

the Los Angeles based users' group, welcomed Dr. Roger Melen, Vice President and Co-founder of Cromemco, as featured speaker at its inaugural meeting for 1984. Cromemcohorts meets the first Tuesday of each month. See **Local Users' Groups** for contact information.

New Products from Cromemco

Over the past few months Cromemco has released a number of new products—both software and hardware—all of which deserve comment.

One capability that has been in demand for some time is answered by Cromemco's **Floating Point Processor (FFP)** board. This card is available for D-Series (68000-based) systems **only.** It is actually a coprocessor which was designed to meet the needs of scientific and engineering computing. The FFP is fully integrated with new "fast" versions of Cromemco's D-Series FORTRAN, Pascal, and C programming languages. All floating point arithmetic is handled according to IEEE standard format in single precision (32 bits), or double precision (64 bits), with major speed improvements over normal DPU machines. **Whetstone** benchmark results indicate that an FFP equipped system is on the order of four times faster than its standard DPU counterpart.

Another product set released is comprised of the **SMDI hard disk controller** and **SMDS software,** which allow users to add fixed or removable media disk drives with storage module (SMD) interfaces. The SMDI can control up to two drives ranging from 8 to 600 megabytes each. That's up to 1.2 gigabytes! A Cromemco system with this capability is performing in a range previously reserved for the better minicomputers. A major benefit of the SMD standard interface is that it allows systems to be configured from a wide selection of vendors and models of disk drives. One other benefit is the speed. The SMDI handles data transfers at a 6 megahertz to 10 megahertz rate with full onboard sector buffering and sector interleaving. Overall, this product set is a very welcome addition to the Cromemco line.

The latest offering, scheduled for release about the time you receive this issue, is the **OCTART.** This dandy card is an intelligent I/O controller which provides complete asynchronous communications support for up to **eight** serial devices. Translated into card slots, the OCTART uses one where there were four TU-ARTS, or where there were one IOP and two QUADARTS. Not only that, but one can add up to four OCTARTS, thus allowing support for up to 32 serial channels. The release of the OCTART will be especially good news for those who have been trying to run more than three terminals under CROMIX. New versions of CROMIX are anticipated to be released through the SUDS program simultaneous with the first shipments of OCTARTS. This card has many other features which make it a welcome product. Check with your local dealers for particulars.

One thing all these new product offerings have in common is that they are only possible because of Cromemco's bus-oriented architecture. The advent of new and powerful technology means newer and better product offerings each year. This is the time to pat yourselves on the back for selecting your S-100

systems.

Floppy Diskettes...
with information on care and handling, as well as how they are made, was the topic of discussion presented by Pat Shaw of 3M for the members of the **North Texas Cromemco Users' Group** at their December meeting. As part of the meeting agenda, information was presented on CROMIX, new developments at Cromemco were discussed, updates on 256KZ diagnostics were explored, and there was a special progress report on the group's program library.

Peter Ingerman...
President of **Microcomputer Users' Group (MUG)**, the innovative local association serving the South Jersey/Philadelphia area, recently headed a discussion on Uninterruptable Power Supplies (UPS). The development of new products in microcomputing and various options for adding hard disks to existing systems were also discussed.

32K Classroom

32K Classroom is a regular column aimed at explaining some programming techniques using 32K Structured BASIC. The main emphasis is on "how to" with sceondary emphasis on coding effectiveness. 32K Classroom is edited by Michael Turnage, President of Turnage & Turnage, Ltd., a software development company in Sunnymead, California.

Mr. Dennis Matthews (member #01469) writes: "...I think many users of 32K BASIC would benefit from a program to read disk labels and which would follow the format of the example on pages 246 and 247 of the Cromemco Basic Instruction Manual. The appropriate Call is CDOS System Call 149 listed on page 87 of the CDOS User's Manual and further described on page 16 of the same Series-2 CDOS instruction manual."

"Cromemco's notes on page 246 of the 32K Basic Manual state that the example on page 247 '...will allow the Basic user to perform CDOS system calls.' This implies a generality to lines 100 through 160 of the example which is not readily obvious to a non-assembly language programmer. If the routine mentioned in the example program really provides this capability, then an explanation of how to implement the program for any System Call would be a logical extension or substitute for my suggestion regarding a program to read disk labels."

Bill Jaenicke, I/O News Technical Editor, looked into the problem and submitted the following explanation and example:

In reading the pertinent parts of the 32K Structured BASIC Manual, I can see how the statement regarding a general method for making CDOS System Calls is a little misleading. The intent, however, was not to say that any system call could be made by simply substituting the desired system call number in the USR function. There can be slight differences due to the differences in the system calls themselves. I will try to explain why this is so.

Not only do each of the CDOS System calls perform quite unique functions, but each has its own requirements as to input and output parameters. Most of the CDOS system calls require that specific information be present in certain registers before the system call is executed. For example, the System Call Number must be present in the BC register pair before the call is invoked by way of the Call 5 instruction. Likewise, the system calls may return information by depositing the information in a specific register.

So before you can execute a CDOS System Call, you must already have passed the appropriate information to the appropriate registers. The various system calls can utilize different registers. This is why the implementation of different system calls can vary to some degree. Different assembly level instructions are required in setting up different registers.

In actual practice, when making CDOS System Calls by way of a 32K Structured BASIC program, you will be dependent on three of the machine level instructions provided in the language. These are PEEK, POKE, and USR instructions.

PEEK, as the name implies, allows you to look at a single byte in RAM memory. The argument of PEEK is the memory location of interest. The format of the function is: PEEK(m) where "m" is a memory location. If "m" is decimal, it can be within the range of 0 and 32767. Hexadecimal numbers range between 0 and FFFF.

A couple of examples using PEEK are:

Print Chr$(Peek(5))

will print the ASCII character corresponding to the value contained in memory location 5.

X = Peek(%0100%)

will assign the value contained in memory location 100H to the variable X. Note that hexadecimal numbers are preceded by and terminated with the percent symbol, "%".

The complement to PEEK is the POKE instruction. It allows you to put a specific byte value at a specific memory location. The format is: POKE m,b where "m" is a valid memory location and "b" is an integer byte value. The same range for "m" applies as with the PEEK function. The range for "b" is between 0 and 255. An example using POKE is:

Poke 1000,255

which would place the value of 255 at the decimal memory location 1000.

The USR function is a little bit more complicated. It allows you to execute an assembly language routine that has been POKED into memory. The format of the instruction is: USR(m,Pl,...,Pn). Here "m represents the memory location of the beginning of the assembly language routine, and "Pl" thru "Pn" represents parameters that are pushed onto the CPU stack.

The USR function is adequately explained in the 32K Structured BASIC Manual, with the exception of one rather important error. The manual says that the parameters are PUSHed onto the stack in order, i.e., that Pl is pushed first, followed by P2, and so on up to Pn. This implies that the first POP instruction will bring back Pn. This is NOT the case. In fact, the parameters are pushed in reverse order, i.e., Pn is pushed first, followed by Pn – l, all the way to Pl being pushed last. So the first POP instruction actually yields Pl. The stack implemented in the CPU is a LIFO (Last In/First Out) stack—not a FIFO stack (First In/First Out). That this is indeed the case can be verified by looking at the example given in the manual: the second POP instruction places the system call number in the BC register pair, and the system call number was passed as the last parameter in the USR function call (the "142" in the system call: Dummy = USR(%0103%,0,142)).

Of course, none of this is going to make much sense unless you are familiar to some extent with assembly level programming. Indeed, if you are not at least somewhat experienced in assembly language programming you should not be at-

tempting to use the machine level instructions. If this is the case, I would recommend that you pick up a book on the subject, such as **Programming the Z-80**, by Dr. Rodney Zaks (SYBEX, Inc., 1980). This is an excellent educational and reference source and a must for every assembly language programmers library.

The general method for executing a CDOS System call through a 32K BASIC program can be broken down into the following steps:

STEP 1:

Get familiar with the CDOS System call, as explained in the CDOS Instruction Manual. This is where you will find what the necessary input and return parameters, if any, are. This is an extremely important consideration, for it will determine the parameters passed as part of the USR function, and will dictate the first few assembly language instructions responsible for taking the parameters off the stack and placing them in the appropriate internal registers.

STEP 2:

Once you know the requirements of the CDOS System call, and which parameters to pass, you can write the assembly language routine to implement the desired system call. The first part of the assembly language routine will be responsible for getting the passed parameters into the appropriate registers. This is done by way of POP instructions that have been coded into your assembly language routine.

STEP 3:

Having written the required assembly language routine you must now manually compile it. In other words, you must convert the mnemonic opcodes into their hexadecimal equivalents. Here is where you will need a reference source to provide you with the byte codes of the various assembly language instructions. Dr. Zak's book is very suitable for this. You will have to take care during this part of the procedure: an incorrect byte value poked into memory can result in mysterious happenings. In particular, be aware that some instructions translate into a single byte, whereas other instructions translate into two or three bytes. Also, when specifying a two byte address (16 bit), the low order byte of the address comes BEFORE the high

order byte of the address, when POKED into memory. At the end of this stage you will have a series of hexadecimal numbers that correspond to the compiled assembly level program and address data.

STEP 4:

Now you can begin writing the 32K BASIC program to implement the system call. The program strategy is to POKE the series of hexadecimal numbers, corresponding to the assembly language routine, into a free space in memory. Since CDOS occupies high memory, and memory from 0-100H, there will be free space starting at 100H and extending to the bottom of Sbasic.com (which is loaded below the bottom of CDOS in high memory).

The example given in the manual gives a starting point of 103H. This is as good as any. The assembly language routine is executed from the BASIC program by way of the USR function. You must supply the starting memory location of the assembly language routine and any necessary calling parameters of the system call as part of the USR function. Since you know how many bytes are going to be POKED, you can compute where the free memory begins after the routine has been POKED. You can use this as free memory to store information generated by the system call. This memory can be accessed after the system call by way of PEEK instructions.

All of this is best illustrated by way of an example. Please refer to the sample 32K BASIC program listing. This program will read a CDOS Disk Label by using the System Call 149: Read Disk Label.

Lines 2000-2300:

This is the assembly language routine necessary to implement the system call to read the Disk Label. The first POP instruction places the address of 32 free bytes of memory into the DE register pair. This is a requirement (calling parameter) of System Call 149. Execution of the system call results in 32 bytes of Disk Label information being written to memory starting at the address stored in the DE register pair. This address is specified as 100H; it is the first parameter of the USR function (line 4100), and therefore is on the top of the stack.

The second POP instruction is responsible for moving the system call number (149 decimal equals 95H) into the BC register pair. As this number is the last parameter passed by way of the USR function it will be the last one popped. That it is placed in the BC register is a requirement of the way that system calls are made: namely, the system call number (decimal or hexadecimal) is put in the BC register pair and the system call executed by way of a CALL 5 instruction.

Program control is returned from the assembly program to the BASIC program by way of the RET instruction. This instruction can be used if, and only if, all parameters that were pushed onto the stack are later popped off. In our case this is true. Otherwise, control can be returned by way of a JP (HL) instruction.

Line 2700:

This line is a DATA statement containing the hexadecimal equivalent of the assembly language routine shown in the REMark statements. It was compiled by looking up the byte codes of the assembly language instructions used. These are as follows:

POP DE	equals	D1 Hex
POP BC	equals	C1 Hex
CALL 5	equals	CD, 5, 0 Hex
RET	equals	C9 Hex

All of the above instructions, with the exception of the CALL 5 instruction, are single byte instructions. The CALL 5 instruction requires three (3) bytes: one for the instruction, and two bytes for the 16 bit address of the location to jump to (location 5H). Note that the low order byte (%0005%) of the two byte address comes before the high order byte (%0000%). The assembly language routine consists of six bytes in total.

Line 2900:

Here, a "0" is poked into the first of the 32 free bytes specified. This is done to specify the "current drive" as being the one to read the label of. Other drives can be specified by placing other values in this byte. Refer to the CDOS Manual and the System Call 149 explanations.

Lines 3300-3600:

This is the FOR - NEXT loop responsible for POKEing the assembly

```
 200   Rem ----------------------------------------------------------------
 300   Rem                           READLBL
 400   Rem
 500   Rem     32K BASIC program to read CDOS disk label using the
 600   Rem     CDOS system Call 149: Read Disk Label.
 700   Rem
 800   Rem ----------------------------------------------------------------
 900   Rem     Routine will be put in memory starting at 103H.
1000   Rem     Disk Label will be stored in memory starting at 110H.
1100   Rem     The first byte at this location will be a 0, specifying the
1200   Rem     current disk drive. The disk label name will be contained
1300   Rem     in bytes 111H thru 119H.
1400   Rem     These addresses are passed as parameters by the USR function,
1500   Rem     which puts them on the CPU stack. The assembly language
1600   Rem     routine "pops" them into the appropriate registers.
1700   Rem ----------------------------------------------------------------
1800   Rem
1900   Rem Assembly language routine:
2000   Rem     pop de          ; DE contains address of 32 free bytes
2100   Rem     pop bc          ; BC contains System Call number
2200   Rem     call 5          ; Call Read Disk Label
2300   Rem     ret             ; Return to BASIC
2400   Rem
2500   Rem ----------------------------------------------------------------
2600   Rem Machine code equivalent of assembly program:
2700   Data %00D1%,%00C1%,%00CD%,%0005%,%0000%,%00C9%
2800   Rem
2900   Rem Specify "current drive" by putting "0" at location 110H.
3000   Poke %0110%,0
3100   Rem Poke assembly program into memory at location 103H:
3200   Integer Byte,Index,Dummy
3300     For Index=%0103% To %0108%
3400     Read Byte
3500     Poke Index,Byte
3600     Next Index
3700   Rem
3800   Rem Perform System Call: memory address of routine = 103H
3900   Rem                      memory address free bytes = 110H
4000   Rem                      CDOS System Call          = 149 (95H)
4100   Dummy=Usr(%0103%,%0110%,%0095%)
4200   Rem
4300   Rem Now the Disk label can be read into a string using PEEK.
4400   Dim Label$(7)
4500     For Index=0 To 7
4600     Label$(Index,-1)=Chr$(Peek(Index+%0111%))
4700     Next Index
4800   Rem print Disk label
4900   @ : @"The Disk Label is: ";Label$
5000   Stop
```

language routine into memory. An index counter is used to specify the starting location in memory (103H), and each of the additional five bytes of memory for the assembly language routine. In each pass of the loop a byte is read from the DATA statement and POKEd into memory at the location specified by the integer variable INDEX. Since the routine occupies 103H to 108H, we have free space for the 32 bytes beginning at 109H. I have used 110H as the starting location for these 32 free bytes.

Line 4100:

This is where the assembly language routine is executed by way of the USR function. Note that the USR instruction is a function, and as such must be assigned to a variable. The USR function is invoked by way of an assignment statement to the variable DUMMY. At the successful completion of the USR function call the disk label information will be stored in the 31 free bytes beginning at memory location 111H.

Lines 4500-4700:

This FOR - NEXT loop is responsible for reading the eight (8) bytes corresponding to the Disk Label. The bytes are read one at a time into the string variable, LABEL. Note that the CHR$ function must be used in conjunction with the PEEK instruction to convert the numeric representation of the characters into printable ASCII characters. The remaining bytes contain other information, such as the date assigned to the disk, and could be read into variables in a similar fashion.

Line 4900:

Here, a simple print statement is used to display the Disk Label that was read. After which, the program is terminated via a STOP instruction.

The program could be embellished. For example, you could allow the user to specify the disk drive to be used. Or, the routine could be incorporated into a larger routine to compare the disk label with a specific name to ensure that the proper diskette has been inserted, before executing some application program. Nevertheless, it should prove to be an adequate example of the procedures involved in making a CDOS System Call by way of a 32K Structured Basic program.

CROMIX — A User's Guide

Continued from front cover

Simply from having used CROMIX for so long, I am now in a position where I understand the system very well. In an effort to help others avoid the arduous learning-in process and its assorted pitfalls—as well as for possible commercial advantage—I have written a book on this powerful operating system entitled, "CROMIX—A User's Guide."

This book is some 55,000 words, arranged in twelve chapters covering all aspects of using and managing a CROMIX multi-user environment to maximum advantage. To give you an idea of the book's organization, I have listed the table of contents:

I am dealing with publishers in Australia for the book, which I hope will be released shortly. The purpose of this note is to enquire whether you see any potential for this book in the U.S. and elsewhere. I would be interested in response from you and other members of the IACU.

Editors Note:

I was very much impressed with Leigh Thomas' latest contribution, **CROMIX — A USER'S GUIDE.** The information is presented in a very readable format, the topics being organized in functional groupings. Groups and file access privileges are covered in a thorough manner, topics that the CROMIX Instruction Manual neglects almost entirely. A good deal of attention was given to methods of configuring various systems, in particular, those with multiple printers and terminals, and how to overcome problems that are likely to be encountered. Perhaps the most interesting, and challenging, sections dealt in the design of command files to facilitate computer operations. It is in this area that Thomas' experience and insight really come through. His examples of using command files to create command files exhibit a depth of understanding, and cleverness, that is inspirational. There is something here for everyone that works with CROMIX, be they novice or expert.

Bill Jaenicke, Technical Editor

QUINTEC

FASTBACK

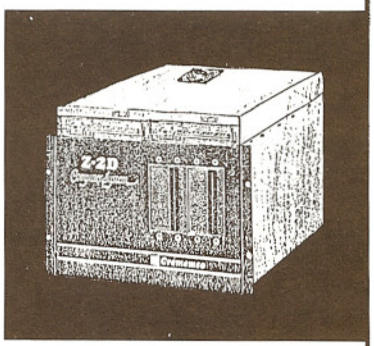
Quintec's well-proven hard disk and floppy backup program will back up 5, 11 or 20 meg hard disks FAST! (11 meg averages 18 minutes on 8" DSDD diskettes

Hundreds sold at $195...Now Only **$149**

QUINTEC TOPHAT

Top off your Cromemco System Two with our 8" floppy disk system. Adds extra cooling to System Twos without adding extra cost. Works with any System Two including Z-2Hs.

Used with FASTBACK, our TOPHAT provides the ideal backup system for all hard disk systems.



- ★ Available in One or Two Disk Drive Setups
- ★ $1395 in Single, $1895 in Doubles
- ★ Brings the AC Power and Reset switches to the Front
- ★ Plugs into Existing System Without Solder
- ★ Uses the Same Tandon TM848-2 Drives as the Cromemco System Three
- ★ Use with 16FDC or 64 FDC Card (specify which)

dLOCK

Our CROMIX utility for Ashton Tate's dBASE II can convert your dBASE programs into truly multi-user applications. Have up to 16 users accessing the SAME data file at the same time without damage to file integrity. Includes:

- Conditional and Unconditional file locking
- Exclusive and Shared file locking
- Up to 16 simultaneous users
- Fast and simple installation

Cost is only **$95.00**

QUINTEC SERVICES, INC.

30313 Canwood St. • Agoura, California 91301

(213) 889-4819

PERSCI REPAIR

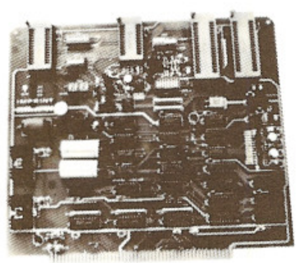
SPECIALISTS IN CROMEMCO/PERSCI
PROFESSIONAL SUPPORT AND DOCUMENTATION
REPAIR SERVICE FROM $175
30-DAY WARRANTY ON SERVICE & PARTS

C-10 GRAPHICS

UTILITIES TO DRAW & PLOT
FOR SBASIC AND PLANMASTER
GRAPH-10 $74.95

COMPUTER CLOSET, INCORPORATED
45F ROUTE 303
VALLEY COTTAGE, NY 10989
(914) 268-5161

Program the Intel Universe with "IMPRINT"



$495

Programs Proms
2708, 2716, 2732, 2732A, 2764, 27128

Programs Microcomputers
8748, 8747H, 8749H, 8751

Program EE Proms
XICOR X2804, X2816

Programs Microcontrollers
8741, 8742

- S-100 Card IEEE-696 extra height to make switches and zero insertion force sockets accessible.
- No extra cables or plug-in modules required.
- Full CP/M™-based software support available.
- Software features Intel fast algorithm.

Inner Access Corporation

P.O. BOX 888 • BELMONT, CA 94002 • (415) 591-8295

Commercial Member Listing

Special Memberships are open to authorized Dealers and OEMs only. These memberships cost $350 per year, and entitle the member to a special listing on the Association's Referral Service Data Base, as well as this printed listing.

North America

Western United States

ACCOUNTABILITY SYSTEMS
700 South Tustin Avenue, Suite B
Orange, CA 92667
(714) 639-4570

An exclusive Cromemco dealership, Accountability Systems caters to the growing business and industrial base in Southern California. The Orange office supports the new personal computer system. Classroom training is available at both locations. CROMIX and Communication specialists. Developers of a professional medical billing package that can be used in single or multi-medical offices. The package provides full accounting for the medical office including monthly Patient Statements, Medicare & Medi-Cal Forms and Standard Insurance. Complete Business Accounting software that is customizable.

Orange Office:
Key Personnel: Michael L. Peterson, Systems Analyst
Kathleen Peterson, Office Manager
Pat McGuire, Jr., Software Systems
Bruce Hughes, CPA, Acctg. Consultant

Glendora office:
PERSONAL COMPUTERS INT'L
104 N. Glendora Avenue
Glendora, CA 91740
(213) 963-3318

Key Personnel: Erroll McGowan, Sales & Marketing
RaeJean McGowan, Office Manager

Major Market Area: Sales & Service in Orange and East Los Angeles counties.
Extended Market Area: Sales & Service in Southern California. Software — Nationwide

AMERICAN COMPUTERS & ENGINEERS, INC.
Corporate Office: 2001 So. Barrington, Suite 204
Los Angeles, CA 90025
(213) 477-6751, Telex: 210-342-6365

Consulting Engineers. Full service company with complete line of hardware and software. Provides to end user on-site or in-house service agreements, fast turnaround on all repairs. Specializing in Structural Engineering, Drafting, Accounting & Word Processing software programs.

Key Personnel: Ghassan Dib, President, (Ph.D.Struct. Eng.)
Aziz Al-Khal, Vice President Sales, (Indust.Eng.)
Brian Gordon, Software Engineer

BRANCH OFFICES:

Orange County, CA: (714) 851-8700

Berkeley, CA: 2855 Telegraph, Suite 210
Berkeley, CA 94705 (415) 849-0177
Key Personnel: Kathy Kolder, Manager

Seattle, WA: 2001 6th Ave., Suite 1612
Seattle, WA 98121 (206) 583-0130
Key Personnel: Bob McNeel, Manager

Spokane, WA: North 9 Post, Suite 241
Spokane, WA 99201 (509) 838-9800
Key Personnel: Jay Townsend, Manager

Paris, FRANCE: 6 Rue Rochambeau 75009
Paris, France Ph: 285-4640
Key Personnel: Maurice Gaspard, Manager

Glasgow, SCOTLAND: 241-A W. George Street
Glasgow, Scotland PH: 248-2427
Key Personnel: Harvey Johnston

Athens, GREECE: 29 Arcadias Street
Athens, 608 Greece PH: 779-8868
Key Personnel: Dennis Ioakim

Major Market Area: Sales and Service, Worldwide

MCM ENTERPRISES
215 Hamilton Ave.
Palo Alto, CA 94301
(415) 327-8080
(415) 493-3333

Sales, Service, Integration, Installation, and Innovation — these are key words to describe MCM Enterprises. MCM is a full service computer solutions company with consulting, equipment, software, training, and service. MCM carries a full line of Cromemco Systems, NEC, Epson & Okidata Printers, Lear Tristar, Peachtree, Micropro (the WordStar People), ProCall Communications Software, and other specialized software. MCM Enterprises also offers full service on NEC Spinwriters, PerSci floppy drives, and all Cromemco equipment. MCM offers a variety of equipment and program service agreements. MCM also custom configures systems for international power requirements and has full export services. Call for training on CDOS, Cromix and languages, as well as hardware.

Key Personnel: M.C. Merchant (MSEE), Owner
T. Bidstrup, Sales
M. Ridgway (MS), Program Service
R. Blaylock, Equipment Service
M. Nadaire (MSEE), Mgr. Paris Office

Major Market Area:
Sales: San Francisco Peninsula & Nevada extending internationally.
Service: S.F. Peninsula and Nevada extending into N. California.
Paris Office: 4 Rue Paul Bert
92150 Suresnes, France
Tel (1) 506 33 03 TLX 610994F

MULTI-MEDIA VIDEO INC.
3350 Scott Blvd., Bldg. 21
Santa Clara, Ca. 95051
Tel: (408) 727-1733
Tlx: 171-577 MMV USA

Multi-Media Video, (MMV), markets bilingual Arabic/English Cromemco systems and peripherals throughout the Middle East. Installations have been made in the government and banking sectors; a complete Arabic banking system was developed for the latter.

Key Personnel: Ursula Burger, President
Michael Laugesen, Sales Manager
Clarissa Clayton, Marketing Manager

Major Market Area: Authorized dealers in Egypt, Saudi Arabia, and Pakistan.

Mid United States

ASGARD COMPUTING EQUIPMENT, INC.
121 West 6th Street
Neillsville, WI 54456
(715) 743-3344

823 5th Street
Menominee, MI 49858
(906) 863-6733

Exclusive Cromemco dealer with long term Cromemco association. Office includes on staff Engineers, Accountants & Chemists.

Key Personnel: James L. Bailey, President
Jerry Hagen, Vice President/P.E.
Ed Baetke, Secretary/Treasurer/Chemist

Major Marketing Area: Upper peninsula in MI, Northern & Western WI, Eastern MI.

COMPUTER CENTERS OF AMERICA, INC.
2200 Southwest Fwy.
Houston, TX 77098
(713) 527-8008

2629 Stemmons Fwy.
Dallas, TX 75207
(214) 638-4477

5401 Mitchelldale
Suite A3
Houston, TX 77092
(713) 957-8787

Complete line of hardware and software. Repair on warranty service. Distribution and OEM division. In-house software house with products in system and applications software. Special export department to service foreign dealers.

Key Personnel: Avery More, President (Sales)
Lee Dixon, Dallas Manager
Race Feirman, Houston Manager
Moti Tenenhaus, Houston Manager

Major Market Area: Texas, Southwest, Mexico, Middle East

JEPSAN GROUP INCORPORATED
4778 Broadmoor, S.E.
Grand Rapids, MI 49508
(616) 698-8700

Jepsan Group is an exclusive Cromemco dealer located in a professional office environment with two demo rooms and four Cromemco Systems for use by customers. Extensive service facilities for all Cromemco hardware, including expertise in PerSci drive and IMI disk repairs. Software consultation and customizing, with specialties in accounting and business applications. Developers of *File Management*.

Key Personnel: Phil Schneider, Pres.
John Nordine, Vice Pres.
David R. Prus, Computer Systems Engineer

Major Market Area: Sales and Service: Western Michigan
Extended Market Area: Service and Software: U.S. and Canada

COMPUTER SALES, INC.
1700 N. Main Street
Racine, Wisconsin 53402
(414) 634-5558

Full-service computer center with retail showroom, service department and extensive training program as well as in-house software development. Carries full line of Cromemco hardware, software, accessories and literature. Custom programming for professionals, retail and manufacturing. Key personnel have strong background in systems analysis and consultation with emphasis on construction, accounting and business applications.

Key Personnel: Fred Cape, President
Ted Witzig, Vice President
Bill Aronin
Fritz Cape

Major Market Area:
Sales & Service: Southeastern Wisconsin
Specialized Systems: Continental United States

TRADEWIND SYSTEMS
Box 96, West Highway 54
Liberal, KS 67901
(316) 624-8111, IN KS 1-800-362-9000

Exclusive Cromemco dealer, specializing in complete business systems. Provides consulting services. Full inventory.

Key Personnel: David Fuller, Store Manager
Ray Cole, System Development
Kevin Elmore, System Development
Randy Hope, Sales
Clark D. Stewart, President
Wayne Stewart, Vice President

Major Market Area: Sales: S.W. Kansas, extending to Colorado, Kansas, Oklahoma, Texas, New Mexico.
Service: S.W. Kansas

SYNERGISTICS INTERNATIONAL LTD.
35 Fountain Square Plaza, Suite 207
Elgin, IL 60120
(312) 695-7775

Full inventory of Cromemco hardware and software. Custom software developed in-house. Vertical market packages available include: Chiropractic Clinics; Architectural Woodwork Job Costing; Social Service Agency Accounting; Auctioneering. Specializing in providing turnkey systems to small and medium sized businesses.

Key Personnel: Jim Knowles, Pres. (Sales)
Gordon Muirhead, Vice Pres. (Software)

Major Market Area: Sales: Chicago and suburbs, extending to entire U.S. and the U.K. Service: Chicago and suburbs.

Eastern United States

BUTLER GRIFFITH GROUP MEMBERS:

ROYAL DATA, INC.
2199 Garden Street
Titusville, FL 32780
(305) 267-1960

ITT Telex 4994013 Royal Da

Key Personnel: Jency Kelly, President & G.M.
Jean C. Kelly, Office Manager
Charles Brossier, Systems Engineer
John Sen, Director/Technical Support

ROYAL DATA, INC.
600 Northlake Boulevard, Suite 100
Altamonte Springs, FL 32701
(305) 830-7140/894-7641

Key Personnel: Wayne Wilson, Sales Manager
Perry Fisch, Technical Support
Don Gilliam, Bus. Systems Specialist

SYSTEMS ATLANTA, INC.
P.O. Box 99
Highway 5, Toonigh Road
Lebanon, GA 30146
(404) 928-0240

Key Personnel: Charley Dobson, President & G.M.
Betty Dobson, Office Manager
Steve Garrison, Director Technical Support
Cliff Geerdes, Systems Engineer
David Swanson, Engineer

Butler Griffith Group is a complete design, installation and support organization, with more than fifty years comprehensive applications experience in process control, telecommunications, graphics and office automation systems. We stock a complete line of Cromemco products. We are also distributors for Ann Arbor Terminals, DataSouth printers, Epson printers and Televideo Terminals. Featuring TCS TOTAL Accounting software, we are also custom design and development experts for difficult or yet unsolved applications.

Major Market Area: Worldwide, with exports to South America, Europe, the Middle East and Canada.

CCS, INC.
A Computer Services Company
733 Third Avenue
New York, NY 10017
(212) 986-7520

Large Cromemco OEM specializing in custom applications on Cromemco Hardware. Full range of services including hardware sales, rentals, long and short term leasing, custom programming and continuous hardware and software support. Specialists in database and large scale financial applications.

Key Personnel: Richard Levey, Vice President
John Ruffo, Vice President

Major Market Area: U.S. and Major cities throughout the world.

COMPUTER CENTER INC.
Corporate Headquarters
253 West 35th Street 12th Floor
New York, NY 10001
(212) 563-7280

31 East 31st Street
New York, NY 10016
(212) 889-8130

480 Lexington Avenue
New York, NY 10017
(212) 697-0200

21 West Street
New York, NY 10006
(212) 697-0200

333 West 57th Street
New York, NY 10019
(212) 307-1900

Complete computer center housing a full line of Cromemco hardware and software. New Corporate Headquarters serving as a control center for four retail stores in the Manhattan area with three additional locations planned offers support in all aspects of customer service including installation, warranty service and education. Large selection of Cromemco software packages as well as custom programming for business and professional applications. Regular C-10 seminars at 21 West Street store.

Key Personnel: Elliott Rabinowitz, Store Mgr.
Edwin Gonzales, Store Mgr.
Mario Giannini, Sales

Major Market Area: Service — Primarily East Coast.
Sales — Worldwide

COMPUTER CLOSET INC.
45F Route 303
Valley Cottage, NY 10989
(914) 268-5161

Complete systems house providing sales, service and support for the full line of Cromemco hardware and software. Provides system planning and design for custom applications in business, education, and professional fields. Regular schedule of seminars and training classes offered.

Key Personnel: Rick Townsend, President

Major Market Area: Sales: Northeast U.S. and East Coast
Service: Continental U.S.

COMPUTER SYSTEM & TECHNOLOGY, INC.
21-55 44th Road
Long Island City, NY 11101
(212) 937-2900/Telex: 910-429418 CSTNY

Cromemco dealer since 1980, we are known for our resourcefulness and good service. We specialize in 68000/UNIX computers, database, and graphic. Among the products we develop are a graphic workstation for CAD application, and the RETAIL CONTROLLER™ that connects the cash register to the computer for inventory control, sales analysis, and cash management. Key personnel have strong background in engineering, financial markets, and import/export trade.

Key Personnel: Mike Fung, President
Fanny Ho, Manager
Salina Ho, Project Leader

Major Market Area: New York, extending to entire U.S., Canada, China, Europe, and South America

CUSTOM COMPUTER SPECIALISTS, INC.
300 Vanderbilt Motor Parkway
Hauppauge, NY 11788
(516) 369-2199

Full service systems house with retail showroom. Full line of Cromemco hardware, software, accessories, and literature. Provides warranty service, diagnostics, consultation, systems analysis, and custom programming. Special management software for attorneys, mass transportation scheduling, reservations, delivery manifests, education, small businesses. School rentals, teacher training.

Key Personnel: Gregory G. Galdi, Pres.

Major Market Area: Sales: Northeast U.S., extending to East Coast
Service: East Coast extending to Continental U.S.

METRO COMPUTER
Division of F.S.L. Corp.
110 Harvard Street
Brookline, MA 02146
(617) 277-5115

A complete service and support dealership committed to a full line of Cromemco products. Services encompass

custom designed hardware and software for individual interfacing needs such as communications applications, multi-vendor configurations, add-ons, and upgrades. All Cromemco systems on display and available for hands-on demonstration, including color graphics system. In addition to on-site warranty repairs, extended maintenance is offered on either an hourly basis or by contract.

A wide range of software and peripherals compatible with Cromemco systems is available. Metro Computer specializes in office productivity, accounting packages and specialty markets. The consulting division of F.S.L. Corporation emphasizes DESIGN AND ENGINEERING for the INDUSTRIAL and DEFENSE area. Our staff of engineers and technicians are standing by for customer support.

Other Lines: Apple, Hewlett Packard

Key Personnel: Frederick S. Lebow, C.E.O. (Engineer)
Dwight Calhoun, Director of Engineering
Robert J. Durante, Vice President of Finance
Melissa Lavers, Sales Manager
James Ritteman, Technical Support (Sales)

Primary Marketing Area: Massachusetts
Extended Marketing Area: New England States

Mexico

SOPORTE ADMINISTRATIO COMPUTACIONAL, S.A.
15 de Mayo 1111 Pte.
Monterrey, N.L., Mexico
Tels. (83) 43-83-40 and 44-62-69

Complete line of Cromemco hardware and software in inventory. Specializing in Business with Software packages (in CROMEMCO Structured Basic) in Spanish according to Mexican Laws and Taxes including Accounting, Payroll, Accounts Receivable and Payable, Inventories, etc., provides full service facility, including technical consulting, custom software, warranty and repair.

Key Personnel: Juan Angel Perez, Director
Luis Ernesto Rodriguez G., Marketing
Delfino Juarez, MSEE Technical Support

Major Market Area: Northeast Mexico

International

Europe

AGRO MARKETING
B Adzije 7/1, 41000 Zagreb
Yugoslavia
41 417-662 Telex: 862-21741 YU AM 2G

Large full-service facility, with complete line of Cromemco products and proprietary software. Specializing in software development, interfacing, and special medical computerized equipment.

Key Personnel: T. Raguz, Director (Marketing)
N. Ivancic, Software Manager
B. Krtolica, Customer Support (Hardware)

Major Market Area:
Sales & Service: Internationally, primarily Yugoslavia

C.T.A. COMBITEXT AUTOMATION
Klein Loolaan 23
3972 KB Friebergen
The Netherlands
03438-18888/Telex: 844-40444 CTA NL

A leading company in the Benelux, in the micro computer market. Represents CROMEMCO computers in these countries. CTA specializes in selling to OEM's, large computer users, and self-programming customers. End-users are supplied with application software via CTA software houses.

Key Personnel: P.H.J.M. Haffmans, Managing Director CTA Int'l.
N. Van Den Bosch, Managing Director CTA Computers
F. Arnolds, General Manager/Software

Major Market Area: The Netherlands

Mediterranean

ALGORITHM LTD.
183 Syggrou Avenue
Athens, Greece
9330551 or 9345858/Telex: 214296 EMVS

Authorized Cromemco dealer which inventories complete line of Cromemco hardware and software. Specializes in applications relating to system development and application software written for civil engineering and shipping, as well as special software packages for data communication and commercial applications.

Key Personnel: Dimitrius Costakis, Managing Director
Dimitrius Pantalatos, Sales Manager
George Natsis, EDP Manager
Kyriakos Regoukos, Systems Manager

Major Market Area: Greece

Middle East

MICRO COMPUTER SYSTEMS MARKETING CENTER
P.O. Box 1446
Jeddah, Saudi Arabia
(966) (2) 651-7707 or 653-0580/Telex 928-403068 MICSYS SJ

Authorized Dealer in Jeddah-Saudi Arabia (Western Zone) for Cromemco sales and maintenance of computers, peripherals, software development and design. Strong Arabic Software development.

Key Personnel: Abdul Rahman H. Attar, General Manager
Issam Al Safadi, Administrative Manager
M. Ali Khan, Marketing Executive

REALTIME ENGINEERING & DATA ANALYSTS
P.O. Box 278
Dhahran Int'l Airport
Dhahran, Saudi Arabia
(966) (3) 8649043/Telex: 928-670480 READAK SJ

P.O. Box 6156
Jeddah
Saudi Arabia
(966) (2) 6531502

Sales and maintenance of computers, peripherals and supplies within the areas of automation, industrial, business and office. Security systems. Strong in developing ARABIC SYSTEMS (hardware and software) and turnkey projects. Large simulators and facsimile.

Key Personnel: A.A. Salamah, Administrative Director
Nasir Jamil, Manager Digital Systems Div.
Ziyad Ismail, Software Design and Development

Major Market Area: CROMEMCO distributor for Middle East (Saudi Arabia, Gulf Emirates, Iraq, Syria, Jordan, Lebanon)

Far East

ASAHI GLASS
Electronics Group
Special Products Marketing Div.
1-2 Marunouchi, 2 Chome
Chiyodaku, Tokyo 100
Japan
781-24616/Telex: 24616 ASAGLAS

Complete line of Cromemco hardware and software in inventory. 700 sq. foot training room. Specializing in O.S. modifications. Full service facility, providing technical consulting as well as warranty repair service.

Key Personnel: Shigeo Satoh, General Manager (systems)
Norimasa Hori, Manager (sales)
Shinichi Watanabe, Tech/software

Major Market Area: Japan

CHINA DATA PROCESSING CENTER
6th Fl., Yu Ming Mansion
7 Roosevelt Road, Section 1
Taipei, Taiwan
Republic of China
02-392-2284/Telex: 785-19844 CDPC CHOWE

One of the largest importer/exporters of computer business/industrial control systems in Taiwan. With several years of computer engineering experience, Tien Sheng provides turnkey basis and reputable service.

Key Personnel: Mr. R. Sheu
Mr. C.K. Cheng
Mr. M.S. Hu

Major Market Area: Taiwan, Republic of China

Australia

INSYSTEMS PTY. LTD.
337 Moray Street
South Melbourne, Victoria 3205
Australia
(03) 690-2899, telex AA30458
84-86 Pacific Highway
St. Leonards, New South Wales 2065
(02) 439-3788

Australia's largest Cromemco suppliers, with staff of 18, providing professional services in all areas of computer implementation.

Key Personnel: Dr. Simon Rosenbaum, Managing Director
Norman Rosenbaum, General Manager
Peter Bell, Victoria Sales Manager
Ian Holland, Software Consultant
David Taylor, Engineer
Garry Vajda, Sydney Manager

Major Market Area: Australia wide. Dealers in QLD and TAS.

MINICOMP SOFTWARE & EDUCATION PTY. LTD.
378 Forest Road
Hurstville, N.S.W. 2220
Australia
02-570 7233 Telex AA23976

Minicomp supports the complete range of Cromemco Hardware, Software and Literature. Minicomp also has an extensive range of application software supported by their experienced staff of programmers and analysts. A full technical and educational program is supplied to all of Minicomps customers.

Key Personnel: Murray Cleworth, Managing Director
Kim Ballestrin, Software Director
Lyn Lyons, Systems Analyst
Bernard Esner, Sales Manager
Ian Robinson, Technical Support
Robert Scottu, Engineering Software

Major Market Area: National Hardware and Software sales and support.

Inside CROMIX

William E. Jaenicke is an independent software consultant and president of SASi (Satellite Accounting Systems, inc.). In addition to the SASi General Ledger, his firm also developed QTS, a timekeeping and time management report package for professionals. He has been working with Cromemco systems for more than four years, including almost three years of experience with CROMIX. Jaenicke holds regular monthly seminars on CROMIX in his Newport Beach, California offices. He can be reached by phone at (714) 955-2220.

Editor's Note:
As is the case with all of the columns that regularly appear in I/O News, users are encouraged to submit their findings regarding the workings of CROMIX, so that they may appear in INSIDE CROMIX for the benefit of all users. This will help ensure that there will always be fresh and rewarding information presented. In this spirit, Robert W. Turner (Charter Member #02067) submitted the following observations regarding the latest (almost) release of 68000 CROMIX version 20.52.

Perhaps the most exciting new feature of the 68000 CROMIX is the RAMDISK. For those of you unfamiliar with the term, I will briefly describe it. The RAMDISK is a driver that enables CROMIX to treat a section of RAM memory as though it were a block storage device, such as a floppy disk or a hard disk. This means that the RAMDISK can be "mounted," and programs and data copied to it. As you all know, disk access (read and write operations) will considerably slow down the operation of an application program. By treating a portion of RAM memory as a disk, this slowing down due to disk access is eliminated. Thus, applications which normally access a disk can be placed in the RAMDISK, and will run at much greater speed. At the completion of the application, any data files that have been modified in the RAMDISK can be copied to the usual block storage device (floppy disk or hard disk). This is a necessary step since RAM memory is volatile, i.e., information is lost when the power is turned off. Mr. Turner writes...

"I recently spoke to one of your staff about the new release of 68000 CROMIX which was recently released by Cromemco. Two of the major new features are some communications utility programs and RAMDISK.

"Though I have not yet tried the new communications software, I have tried RAMDISK, and it's great! I want to pass along some important information to users who may have trouble getting it to work the first time. RAMDISK must be selected by reconfiguring the operating system using the CROGEN68 utility. Up to 4 separate devices may be specified as RAMDISK, and the user can configure each RAMDISK device to practically any size in 64K byte increments (0 to 250 blocks of 64K bytes for each device). The trick is in getting the necessary device drivers properly loaded in by CROGEN68. If you don't request device drivers for TU-ARTS, QUADARTS or IOP memory, you don't get the device drivers for RAMDISK. I found it necessary to request all of those device drivers (even though I don't have a QUADART or IOP board in my system) so that when I finally requested the RAMDISK device drivers (one of the last requests in the sequence) I got them. My guess is that they must be skipped over during the link load sequence unless other preceding device drivers are also requested.

"The documentation accompanying this release gives a terse example of how to make use of the RAMDISK via a "/ram" directory which the operating system has been programmed to search first before going to the "/bin" or "/cmd" directories. Another command file may be included in "/etc/startup.cmd" file to have the system load the desired files into the "/ram" upon booting up the system. I have taken this two steps further. I have two other RAMDISK devices also mounted: one I use as a working file area, the other I use as the "/usr/spool" directory. The latter is only a little tricky. To make things clear, I have included a listing of my startup sequence. Using RAMDISK for the "/usr/spool" directory speeds up the time required for spooling, and doesn't tie up valuable disk space (I currently am working with the PerSci 299B dual drives)."

/etc/startup.cmd

```
time; time -s
mode timer c 32767
ty /etc/startup.msg
if -r /etc/.warning ty /etc/.warning
ren /etc/warning /etc/.warning >*/dev/null
ram0_load;ram1_load;spool_load
flush 300&
```

/cmd/ram0_load.cmd

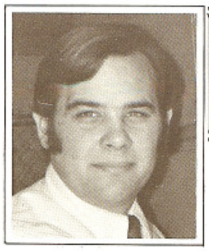
```
%
% procedure to load utility programs and command files into RAMDISK
%       note: DO NOT INCLUDE INIT.COM
%
echo
echo "          Creating /ram directory - [RAMDISK on device: rd0] ..."
echo
% create /ram >* /dev/null
makfs -i 128 rd0
if -err exit
mount rd0 /ram
%
d /bin
copy copy.bin /ram
copy access.bin blink.bin ccall.bin cdoscopy.bin chowner.bin /ram
copy cptree.bin deltree.bin dump.bin echo.bin /ram
copy fortran.com help.bin /ram
copy l.bin ls.bin mail.bin makfs.bin maklink.bin mcsterm.bin /ram
copy mode.bin mount.bin pr.com rfile.bin /ram
copy screen.bin sfile.bin sim.bin sort.bin spool.bin /ram
copy time.bin unmount.bin usage.bin version.bin /ram
%
d /cmd
copy bell.cmd clearprt.cmd ftn.cmd ftnts.cmd getfor.cmd /ram
copy load.cmd loadup.cmd log.cmd print.cmd route.cmd /ram
copy sortdir.cmd sysdir.cmd /ram
copy term1200.cmd term300.cmd /ram
%
d /ram
chowner bin *.bin
chowner system *.cmd
chowner system cptree.bin fortran.com mail.bin mcsterm.bin
chowner system pr.com sim.bin version.bin
maklink help.bin h.bin
access rewa.e.e *.bin
%
echo;usage;echo
```

/cmd/ram1_load.cmd

```
%
% procedure to initialize scratch directory in RAMDISK
%
echo
echo "          Creating /ram1 directory - [RAMDISK on device: rd1] ..."
echo
% create /ram1 >* /dev/null
makfs -i 96 rd1
if -err exit
mount rd1 /ram1
%
d /ram1
%
echo;usage;echo
```

/cmd/spool_load.cmd

```
%
% procedure to initialize print spool directory in RAMDISK
%
echo
echo "      Creating /usr/spool directory - [RAMDISK on device: rd2] ..."
echo
%
makfs -i 32 rd2
if -err exit
mount rd2 /usr/spool
chowner bin /usr/spool
%
d /usr/spool
copy /etc/sequence_number sequence_number
chowner bin sequence_number
%
echo;usage;echo
```



tec·tips

TEC TIPS is a regular column aimed at providing hints for keeping systems up and running. It will not attempt to deal with specific engineering applications or non-standard configurations. TEC TIPS is edited by Richard Quinn, owner of QUINTEC, a Southern California Computer service firm.

Editor's Note:
This issue's **Tec Tips** column represents a slight departure from the usual format. What we have here is an in depth study regarding how to configure a system with multiple printers under CROMIX. The guest editorial is provided by Michael Hazen, Vice-President and Systems Analyst of QUINTEC SERVICES, INC.

MULTIPLE SYSTEM PRINTERS IN CROMIX

Good news for CROMIX users! You really can use several printers in your CROMIX system! You can use them one at a time or even use them all at the same time, and not have them interfere with each other. Let's see how, but first let's look at an obstacle we must overcome.

Under CROMIX, there is a printer that all of your programs print to. This is called the system printer and is named '/dev/prt'. It is actually a dummy name and only has meaning after it is linked to an existing printer driver (a small CROMIX program whose sole responsibility is to provide access to a printer). That is, to select a printer, you must make a link to /dev/prt from the existing printer driver representing the printer you want to use. That printer then becomes the system printer.

Here's the major obstacle to using multiple printers in CROMIX: since there can only be one file at any one time named /dev/prt, there can only be one system printer at any one time in the standard CROMIX system.

There are at least two ways to get around this. The easiest is to link different printer drivers at different times to /dev/prt. We still have only one system printer at a time, but we can select any one printer at any one time to be the system printer. Another way is to allow for multiple system printers at the same time. A working example of this will be the focus of the remainder of this article.

The first need is to establish system printer names. As will be seen later, the printer drivers' names are not appropriate, so others must be created. The following system printer names will do: the first system printer could be called /dev/prl, the second /dev/pr2, and so on. If more than nine are needed, you could call the tenth /dev/pra, the eleventh /dev/prb, and so on.

To set up your system to use these new system printers, you simply use the **maklink** command to link your existing printer drivers to the system printer names you choose. As an example, we use our dot matrix printer as our first system printer, our 3355 letter quality printer as our second, and our serial line printer as our third. The actual CROMIX instructions used to link these are:

```
# maklink /dev/1pt1 /dev/pr1
# maklink /dev/typ1 /dev/pr2
# maklink /dev/s1pt1 /dev/pr3
```

Once you decide which of your printers should be /dev/pr1, /dev/pr2, and so on, you can make your appropriate links and you will be ready to use your new system printers. All you need are programs which use these new printer names.

We have to start somewhere, so let's modify the screen editor to allow it to print to two different printers. By making copies of /bin/screen.bin to /bin/screen1.bin and /bin/screen2.bin, you have working copies in which you can make the changes. You might say:

```
# copy /bin/screen.bin
  /bin/screen1.bin
# copy /bin/screen.bin
  /bin/screen2.bin
```

In each copy, there is exactly one reference to /dev/prt (i.e., the name only shows up once, and it's in lower case). Use **Patch** or **Debug** to change the reference in /bin/screen1.bin to /dev/pr1. Do the same with /dev/pr2. Now, when you use the 'List' command in the screen editor, instead of printing to /dev/prt, it will print to /dev/pr1 for screen 1 and /dev/pr2 for screen2.

Note that the space in each program for the printer reference is not long enough to be replaced by the original printer names (i.e., 1pt1, typ1, or s1pt1). That is why we had to agree on some new system printer names that were short enough to fit into the spaces provided.

It's now easy to use the screen editor to list to any printer. Instead of typing:

```
# screen letter
```

to edit a file named 'letter' and list to the system printer, type:

```
# screen1 letter
```

and list to the first system printer, or

```
# screen2 letter
```

and list to the second system printer. So, with minimal cooperation, several people in your office can be editing and printing to specific printers at the same time without fear of conflicts.

If conflicts do arise, you may need to resolve them with the **Spool** command. This program allows several requests for a printer at one time without conflicts. To spool the file 'letter' to the first system printer, you can type:

```
# spool /dev/pr1 letter
```

You can do the same for the other system printers, without fear of disrupting any other print request(s).

That was easy; let's try something a little harder, like the CDOS Simulator. It's the program CROMIX uses to run all of your CP/M and CDOS programs (like BASIC, WordStar, dBASE II, etc.). When CROMIX determines that it has been asked to run a CDOS or CP/M program, it first loads the simulator (/bin/sim.bin) and passes all the appropriate parameters to the simulator to

allow it to load and execute the program.

When the program which the simulator is executing asks to print something, the original simulator converts all print requests for the CP/M list device (the printer) to requests for /dev/prt. By modifying the simulator in a way similar to the screen editor modification above, the modified simulators can convert the references to /dev/pr1, /dev/pr2, and so on.

To create simulators for the first two system printers, first make a couple of working copies:

 # copy /bin/sim.bin /bin/sim.pr1
 # copy /bin/sim.bin /bin/sim.pr2

Once again, there is exactly one reference to /dev/prt in the original simulator. Now use **Patch** or **Debug** to change the reference in sim.pr1 from /dev/prt to /dev/pr1, and the reference in sim.pr2 to /dev/pr2, and you have a simulator for each of the first two system printers.

Now, if you are in a directory with either a CP/M or CDOS program in it and you wish to run it and print to a specific printer, you could first select the printer by selecting the appropriate simulator. Let's use an example to explain the process. Suppose you are in a directory with dBASE II in it and you wish to run a dBASE II command file named **menu.cmd** and send all printer output from it to the second system printer. You could use the following commands:

 # maklink /bin/sim.pr2 sim.bin
 # sim dbase.com menu

The first command puts the appropriate simulator into the current directory and gives it the name CROMIX requires it to have (yes, CROMIX requires 'sim.bin'; nothing else will do). The second command calls up this simulator, tells it to run dBASE II and tells dBASE II to run menu.cmd.

If you always use at most one modified simulator when in a specific directory, you could feasibly leave the modified simulator in the directory and just use the second command to run your programs. However, this may not always be the case (i.e., you may wish to print to one printer sometimes and another at other times). If not, you may want to delete the modified simulator in the current directory:

 # del sim.bin

before continuing processing, or use the force option when linking the next time:

 # maklink -f /bin/sim.pr2 sim.bin

The problem arises that more than one person may be in a directory and calling up a modified simulator at any one time. The **maklink** command will tell you if a file named sim.bin is already in the current directory (if the 'f' option is not used), and if so, you may decide not to continue. There are several options you have for setting up a procedure for your system. An example procedure follows.

First, only one person may use a modified simulator in a directory at any one time. If this poses a problem, additional directories are set up using the **maklink** command. Next, the commands to call up the software with a modified simulator are handled with CROMIX command files. An example WordStar command file follows, named /cmd/ws3.cd:

% Command to run Wordstar with the 3rd system printer
maklink /bin/sim.pr3 sim.bin
if **-err exit**
sim ws.com #*
del sim.bin

The 'if' statement exits the command file if a simulator is already in the current directory. The '#*' passes all parameters to WordStar. That is, to call up this command file and edit a file called letter, we can type:

 # ws3 letter

CROMIX command files unfortunately use up system buffers and if several command files are running at the same time, CROMIX may give you the 'No system buffers available' message, which defeats the use of command files. If there is a recurring problem at your facility, we suggest typing out the commands from the CROMIX shell instead of using command files. If the maklink command tells you that 'sim.bin' already exists in the current directory, you would probably not want to proceed until you have determined that you are in the correct directory and that the 'sim.bin' file in it is the one you need.

One problem arises with this procedure when the program is not properly exited (e.g. someone else locks up the system or you have a power outage while executing a command file similar to the one above): the modified simulator is left in the current directory and all future attempts to run the command file fail because 'File already exists: "sim.bin"'. In such a case, you may delete sim.bin in the current directory and execute the command file again.

You may also want to spool from a CDOS or CP/M program instead of assuming that a particular printer is available. Some programs (e.g. dBASE II) have utilities available to allow spooling when printing, but many don't. You can create another modified simulator for this purpose. Let's call it /bin/sim.spl (for spool). In this simulator, change the reference from **/dev/prt** to **spoolfil** (spooled file in the current directory, i.e., not /spoolfil, or /dev/spoolfil, but just spoolfil). If this name conflicts with any in your directory, you can settle on a universal name and use it instead. To make the setup easiest, use an eight-letter name.

Now when you are in your desired directory, /bin/sim.spl is linked into the current directory as sim.bin and you are using it to execute a program. Instead of printing directly, your program will overwrite any existing file called 'spoolfil', appending all print information to the file (including form feeds, bold print and underline commands, along with text, for each time you have requested your program to print). Exit the program and print the file using the **spool** command and voila, you are effectively spooling from your CP/M or CDOS program. A little roundabout, but functional.

That's about it. As you can see, CROMIX has the power and flexibility, and you have the tenacity (right?) to make use of every printer in your facility which you can attach to your system. A little determination and creativity can add a lot of power to your CROMIX system!

Soft Tips

SOFT TIPS is a regular column aimed at providing software oriented hints and ideas for non-programmers. Members are encouraged to send in tips that can help a user better use his/her system. SOFT TIPS is designed to put forth ideas that are general in nature. The column is edited by Norman Vadnais, President of **Computer Specialists & Associates**, an Orange County Customer Support Specialist. Member's contributions can be sent to SOFT TIPS in care of I/O News.

LETTING CROMIX RUN YOUR SYSTEM

Many people have discussed various methods for defeating the software handling of the printer by CROMIX. The many modes of CROMIX can be easily modified by the user and CROMIX offers many different accesses to the printer other than the spooler. By allowing the printer to be controlled by CROMIX, a very efficient environment can be obtained.

The main reason behind defeat of the simulator is often the use of CDOS or CP/M programs that are not capable of activating the spooler. Though this is true of those using canned programs, other programs, including languages like 32K Structured BASIC, can be modified to use the spooler. Even if programs not using the spooler are used, the spooler should be used to maintain page control (i.e., control top of form). The TOF command file (I/O NEWS, Volume 2, Number 5) was designed for just this purpose.

The rules to remember when using CDOS or CP/M programs that access the printer directly are: 1) Make sure the proper form length is set in CROMIX, if your forms change fix this setting; 2) CROMIX is always counting the lines output to the printer, if your program has form control, CROMIX's bottom margin should be zero; 3) The printer is unaccessible to everyone else as long as a program that used the printer is still running; 4) If an operator manually skips lines or pages on the printer, there is the possibility that CROMIX will try to skip a page in the middle of a printed page.

SYSTEMS PROGRAMMERS CORNER

For those of you who dabble in assembly programming, the System Programmer Corner will try to keep you informed and offer you a forum to discuss your findings. This section, however, will only offer operating system type features and hints, versus the latest sorting algorithm. If you have any ideas to contribute to the SPC, please forward them to SOFT TIPS, in care of I/O NEWS.

When you made the step from CDOS to CROMIX, you may have noticed that there was no longer a rename file system call. Most of you have probably realized a solution, but for those who have not, here goes. The simplest way to rename a file with CROMIX system calls is through the use of the **.f link** and **.delete** system calls. If a rename is desired, the registers as returned by the **.f link** system call are already set for the **.delete** system call. By setting the register to point to the new and old names for the **.flink** call, the registers have been set to perform the delete.

Another item to note, forwarded by Rick Dhaenens, Technical Support Manager at Cromemco-Atlanta, has to do with the HDTEST program. A bug has been noted affecting all users running DPU based systems, no matter which operating system is being used. The HDTEST program, on a system with a DPU, will report the RPM of all hard disks approximately 13% higher than the actual RPM. At the moment, this bug is not a high priority at Cromemco, so please note this bug and pass it on to anyone with a DPU-based system.

CDOS DRIVES ON CROMIX SYSTEMS

As has been seen in many of the recent issues of I/O NEWS, there are many ways in which CDOS drive designators can be optimized under CROMIX. I hope I can highlight some of the points made in these recent articles and add a few of my own.

As seen in this column, Volume 3, Number 1, it is possible to make CROMIX look for CDOS drives B-H at a variety of places. As shown by Robert Brown, Jr., a common location for the drives is as subdirectories to a main directory. There is one part of this scheme that I have received many questions on: where do you put your main programs, the ones looking to the additional drives? The programs requesting the additional drives must also be in a subdirectory to that same main directory. Please note, you can use alternate path names for your alternate directories. Basically, any pathname of one or two acceptable CROMIX characters is allowed, followed by the required slash.

In Volume 3, Number 2 of I/O NEWS, Peter Zadrozny provided some excellent "patch" work for many different programs. The most useful items offered are the patches for canned CP/M software packages requiring overlays. This process is available for most canned programs. The bytes to change, however, are sometimes hard to find. Anyone wishing to share patches they have come across are asked to forward them to SOFT TIPS, c/o I/O NEWS.

Some patch locations are offered here for those who use Cromemco's COBOL. For Cromemco's COBOL, the following patch allows the four overlay programs, coboIX.ovr, where X is a 1, 2, 3, or 4, to be situated on another drive. This patch can be useful on a CDOS or CROMIX system to specify drives B through H. The main program, cobol.com, needs to be patched in one location to effect the changes, 6D61h for debug or 6C61h for patch. The following commands show the patching (here using debug):

```
# debug cobol.com
DEBUG version 00.20
NEXT = 7C00
NEXTM = 7C00
– s 6d61
6D61 00      8      (2 for drive B,
   3 for drive C, etc. Here, we put
```

an 8 for drive H or the /H directory)

6D62 43

– w

– ^ (CONTROL C to quit)

HANDY COMMAND FILES

We have, in the past few issues, discussed CROMIX's ability to make decisions through the IF command. We have discussed the many different ways that IF can make decisions on items presented in a command line. But what about decisions based on user inputs? The IF command does not directly address this situation. There are two commands, however, that combine to fill the need for decisions based on user inputs: **input** and **testinp.**

INPUT accepts input from the terminal and stores it as requested by the command line. TESTINP then compares the input to some preset data and sets the CROMIX error flag when appropriate. The error condition mode of the IF command is then used to make our decision.

The proper syntax for the input command is: **input.** The program will read input from the terminal until it receives a carriage return or line feed and then place the input where instructed. If no instructions are given, the input is echoed back to the terminal. The power of the input command comes from the ability to redirect its output. The most commonly used syntax of input is: **input > your__file.** The greater than symbol, ">", directs the location of the output. It may be any file the user desires. The file can then be tested with the **testinp** command.

The proper syntax for the **testinp** command is: **testinp [options] test__file string1 string2 ....** Test__file is the file containing the item to be tested and is usually the file listed in a previous input command. The strings, string1 and so forth, are the items used for comparison. The strings are compared character for character with the test__file without any regard for upper or lower case. If the contents of test__file do not match any of the strings, the CROMIX error condition is set. The error detect mode of the CROMIX 'if' command is then used to complete the decision making process.

There are three options to the **testinp** command, all of which are placed behind a single dash (-) when any or all options are requested. The most common option is the **d** option, which will delete test__file after the comparison. Command files which use this command process will almost always use this option to delete the temporary storage file. The second option is the **f** option, which causes only the first characters of both test__file and the strings compared rather than the full string. The last option is the **r** or reverse option, which reverses the effect of the test. The CROMIX error condition will be set for all matches and will not be set if there is not a match.

The error detect mode of the CROMIX "if" command tests the error condition left by **testinp.** Most times, the "if" command conditionally executes the "goto" command to skip to the proper command steps. A second and very common usage is to abort command files per user request, especially command files altering the system. Along with the example presented here, the current CROMIX manual, page 249, offers an example of the input/testinp procedure.

Next time around we will try to discuss user questions regarding command files we have printed previously, and explain some of the simpler terminology used in these columns to date.

Name: **GAMES**

Purpose: Offers a menu driven game library accessible to all users. Can be renamed and offer a menu driven system of any other set of programs.

Setup: A text file, here named games.menu, must be created to clear the screen, offer the user his choices, and prompt for his response at the end of the screen.

Listing:

```
%display__menu
ty games.menu
input > games.temp
testing –r games.temp 0
if –err goto done
testing –r games.temp 1
if –err football
testing –r games.temp 2
if —err tictctoe
testing –r games.temp3
if –err life
testing –r games.temp 4
if –err 23 match
testinp games.temp 1 2 3 4
if –err /bin/echo –e '^G'
goto display__menu
%done
/bin/echo –n '^Z'
del games.temp
```

Notes:

The display__menu section of the command file displays the options available to the user and prompts for the user's choice of game (our options are 1 for football, 2 for tic-tac-toe, 3 for life, 4 for 23 matches, and 0 to exit). The **input** command is then used to place the user's response into a file called games.temp. The potential responses are then compared one by one. The first response tested for is 0. As with all of our response tests, the **–r** option of the **testinp** command is used to set CROMIX's error condition if the test character matches the data file. In this case, if the character was a zero, the command file jumps to the done area. This will echo a clear screen command (here we use the clear character for a Televideo terminal) and delete the temporary file. If our input was not a zero, all other responses are tested for in order, with the corresponding game being executed when a match occurs. The final test is performed to verify proper input. It tests to verify that a valid response has been made. If a valid response was not found, this test will cause a beep to be echoed to the console. After testing for all possible responses, the command file returns to the display__menu section and starts over. Please note, this method becomes very inefficient when the number of choices becomes large.

Example:

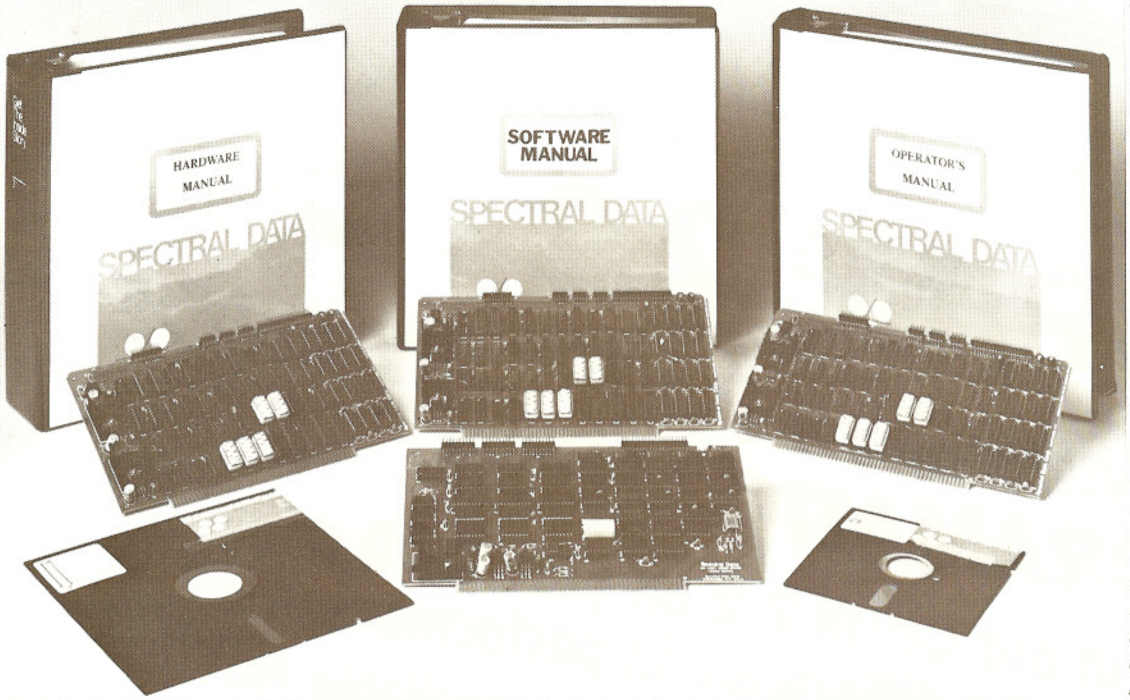
% **games**	Your options are:
Your options are:	1. Football
1. Football	2. Tic-tac-toe
2. Tic-tac-toe	3. Life
3. Life	4. 23 matches
4. 23 matches	0. Exit
0. Exit	Your choice: **0**
Your choice: **2**	(((the screen is % cleared)))
(((a game of tic-tac-toe is played)))	

We're interested!

We're Really Interested...in what you have to say. Especially about how you use your system... the problems encountered and the solutions effected... unusual uses or environments... and any practical applications you would be willing to share with fellow members. These can be short notes for departments like 'bits & bytes...' and 'Tec Tips', or full feature articles.

Contact Richard Kaye or Lynn Platzek at I/O News for editorial guidelines or assistance. We're interested in unleashing your literary talents.

IMAGE PROCESSING FOR CROMEMCO OWNERS! $3,800



DIMENSION: Spectral Data Corporation offers a new dimension in image processing.

BEGIN: With a Cromemco System 2 or System 3 running with CDOS version ≥ 2.52 (also CROMIX)

IF: You want to access the world of Image Processing and want to chose from a world of Satellite Images; then CALL SPECTRAL DATA.

DETAILS: Spectral Data's Video Image System consists of three 32K RAM boards capable of displaying 256 x 240 pixels x 4096 colors

Video boards allow the user to run all of the U.S. government generated RIPS programs consisting of 65 program modules.

Government Satellite imagery available on 8 inch Floppy Disks.

Spectral Data image processing packages available.

Color Printer hardware/software available.

Digitizer hardware/software available.

APPLICATIONS: Digital Analysis of LandSat and Aerial Photographs

Mineral and Petroleum exploration

Enhancement of medical radiographic images

Inventory and Assessment of Forest Resources

Water Resources Assessment

Land Use Management

Assessment of Population Distribution and Growth

Military Reconnaissance.

ADVANTAGES: To the Cromemco User are substantial; For it is ABSolutely the least expensive way to ENTER the growing field of Image Processing.

Spectral Data CORPORATION

112 PARKWAY DRIVE SO., HAUPPAUGE, NY 11788

TELEX/TWX 510-226-1481 (SPECTRAL HAUP)

CALL 516-543-4441

Please Print

Name ______________________

Company ______________________

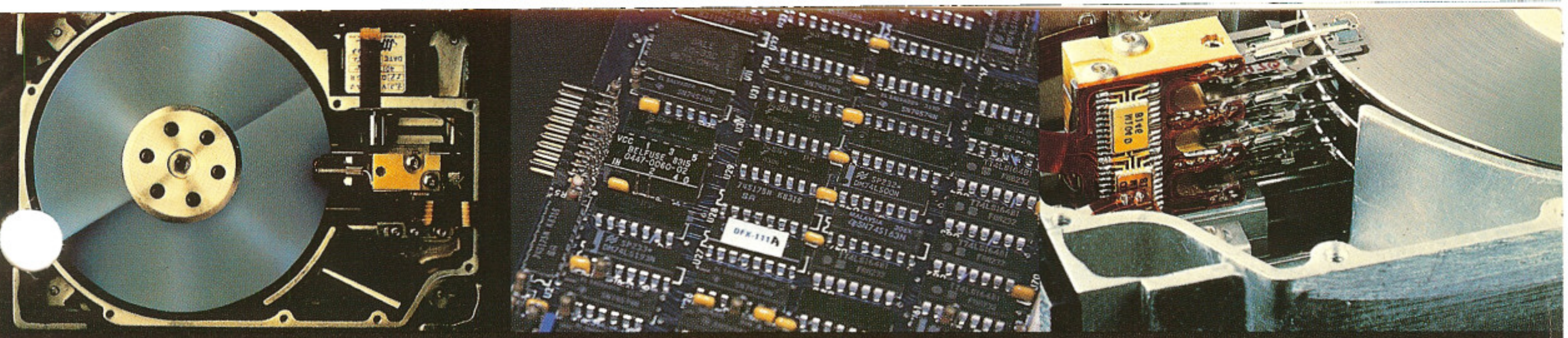
Address ______________________

City ____________ **State/Zip** ____________

Phone # ______________________

☐ Video Board Upgrade.

☐ RIPS mainframe and accessories.



How do you spell relief?

AUXIN™ The ultimate mass storage relief machine for your Cromemco computer.

With up to 18 megabytes (300 Mb on EXOCET™ models) of hard disk capacity, you can handle extremely large data bases. Simply access, process, and update all that data in micro-seconds.

Which, after all, is what serious personal/business computing is about.

PERFORMANCE.

AUXIN series hard disks are truly beautiful engineering marvels.

FAST.

Data moves swiftly in and out at 5 million bits per second.

QUIET.

Even with the disks spinning at 3600 rpm inside.

SOLID.

Which is why AUXIN series drives carry a warranty few others would dare offer. 2 years on disk drive, 12 months on controller and chassis, including parts and labor.

EASE OF USE.

Just plug the AUXIN drive controller into your System One, System Two, System Three, or even C-10, and you are ready to go.

AUXIN drives come with plain English instructions and the software required to make them perform marvels with CROMIX or CDOS. CP/M Plus will be available early 1984.

LUXURY AT ECONOMY PRICE.

The best way to order an AUXIN series drive is right from your local dealer, or directly from the factory. You pay considerably less than you would for a lesser system sold elsewhere. Powerful standalone AUXIN mass storage systems start at $2495.

AUXIN 18c-18 Mb Hard Disk for C-10	$2,495
AUXIN 18s-18 Mb Hard Disk for CS1, CS2, CS3,	$2,495
AUXIN 18a-18 Mb Hard Disk Add-on	$1,995
EXOCET 315-300 Mb Hard Disk for CROMIX Systems	$12,995

Call or write for information on other products, models and/or special configurations available.

To reach our professional sales staff, dial (415) 366-4610.

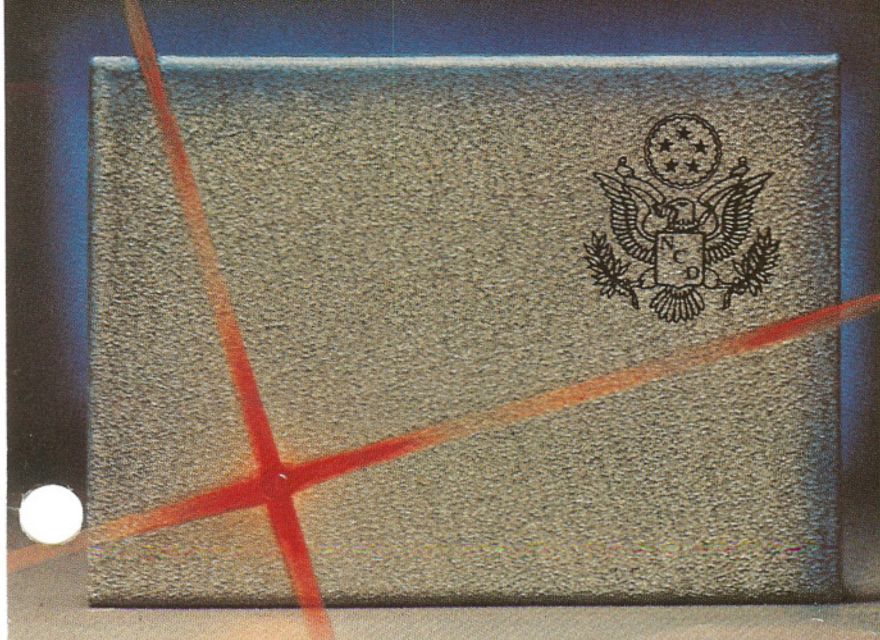
You may charge the AUXIN to your VISA or MasterCard or you may send a check, money order or wire transfer. Your AUXIN mass storage system will arrive in virtually days.

And your Cromemco will get a much needed boost in mass storage horsepower.

RELIEF.

Auxin's high performance, low cost, extended warranty, and immediate availability.

Dealer Inquiries Invited



National Cryptographic Devices, Inc.
Peripheral Division
635 Bair Island, P.O. Box 2938
Redwood City, CA. 94064
(415) 366-4610

How Cromemco plugs you into the state of the art.

Cromemco offers you the most complete line of S-100 boards and peripherals in the business. These boards use the new IEEE-696 state-of-the-art standard. One-stop shopping can satisfy your design needs the easy way.

You can build one system, or a hundred, exactly the way you want, and upgrade existing systems with a simple board swap or addition. And since we design our own boards for our own systems, we always take advantage of the latest developments in IC technology.

68000 microprocessor performance. Cromemco's Dual Processor Unit gives you the best of both worlds: the 68000 and the Z-80A microprocessors on the same board. It's the easiest way to move into 68000 performance and still use your existing 8-bit software. Or use Cromemco's Z-80A CPU board or our Z-80A-based single board computer.

For selection, Cromemco can't be matched. From the well-known SDI High Resolution Color Graphics board to the new 512MSU, 512K byte RAM board. From our highly reliable Local Area Network interface (C-NET) to our wide variety of general purpose interface boards. And you can put them in one of our 8-, 12-, or 21-slot card cages with our 12 amp PS-8 power supply to get your system into operation fast.

We have over 30 S-100 boards to fill your needs. And all are supported by a broad line of software. Our Board Products Catalog has the latest information. Call today for your copy, or to get the name of our nearest dealer or distributor. Or, write Cromemco, Inc., 280 Bernardo Avenue, P.O. Box 7400, Mountain View, CA 94039. Tel: (415) 964-7400. In Europe: Cromemco A/S, Vesterbrogade 1C, 1620 Copenhagen, Denmark.

In the U.S., contact your local Hallmark or Kierulff distributor.

Cromemco